Learning Gaussian graphical models with fractional marginal pseudo-likelihood

Leppä-Aho, Janne

2017-01-11

Leppä-Aho , J , Pensar , J , Roos , T T & Corander , J 2017 , ' Learning Gaussian graphical models with fractional marginal pseudo-likelihood ' , International Journal of Approximate Reasoning , vol. 83 , pp. 21-42 . https://doi.org/10.1016/j.ijar.2017.01.001

http://hdl.handle.net/10138/297770
https://doi.org/10.1016/j.ijar.2017.01.001

International Journal of Approximate Reasoning (2017) DOI: 10.1016/j.ijar.2017.01.001

Learning Gaussian Graphical Models With Fractional Marginal Pseudo-likelihood

Janne Leppä-aho[a], Johan Pensar[b], Teemu Roos[a], Jukka Corander[c,d]

[a]*HIIT / Department of Computer Science, University of Helsinki, Gustaf Hällströmin katu 2, 00560 Helsinki, Finland*
[b]*Department of Mathematics and Statistics, Åbo Akademi University, Vänrikinkatu 3, 20500 Turku, Finland*
[c]*Department of Biostatistics, University of Oslo, Sognsvannsveien 9, 0372 Oslo, Norway*
[d]*Department of Mathematics and Statistics, University of Helsinki, Gustaf Hällströmin katu 2, 00560 Helsinki, Finland*

Abstract

We propose a Bayesian approximate inference method for learning the dependence structure of a Gaussian graphical model. Using pseudo-likelihood, we derive an analytical expression to approximate the marginal likelihood for an arbitrary graph structure without invoking any assumptions about decomposability. The majority of the existing methods for learning Gaussian graphical models are either restricted to decomposable graphs or require specification of a tuning parameter that may have a substantial impact on learned structures. By combining a simple sparsity inducing prior for the graph structures with a default reference prior for the model parameters, we obtain a fast and easily applicable scoring function that works well for even high-dimensional data. We demonstrate the favourable performance of our approach by large-scale comparisons against the leading methods for learning non-decomposable Gaussian graphical models. A theoretical justification for our method is provided by showing that it yields a consistent estimator of the graph structure.

Keywords: Approximate likelihood, Fractional Bayes factors, Model selection, Structure learning, Gaussian graphical models

1. Introduction

1.1. Bayesian learning of Gaussian graphical models

Gaussian graphical models provide a convenient framework for analysing conditional independence in continuous multivariate systems [1, 2, 3]. We consider the problem of learning Gaussian graphical models from data using a Bayesian approach. Most of the Bayesian methods for learning Gaussian graphical models make the assumption about the decomposability of the underlying graph [4, 5, 6]. Recently, Fitch et al. [7] investigated how Bayesian methods assuming decomposability perform in model selection when the true underlying model is non-decomposable. Bayesian methods that do not assume decomposability have been considered more seldom in the literature, and in particular not in the high-dimensional case [8, 9, 10, 11, 12, 13, 14].

A widely used frequentist method for learning Gaussian graphical models is the graphical lasso [15, 16]. Graphical lasso (`glasso`) uses l_1-penalized Gaussian log-likelihood to estimate the inverse covariance matrices and does not rely on the assumption of decomposability. Other approaches include a neighbourhood selection (`NBS`) method by Meinshausen and Bühlmann [17] and Sparse Partial Correlation Estimation method (`space`) by Peng et al. [18]. The `NBS`-method estimates the graphical structure by performing independent lasso regressions for each variable to find the estimates for the neighbourhoods whereas `space` imposes an l_1-penalty on an objective function corresponding to

Email address: janne.leppa-aho@helsinki.fi (Janne Leppä-aho)

an l_2-loss of a regression problem in order to estimate the non-zero partial correlations which correspond to edges in the graphical model.

Assuming decomposability in Bayesian methods has been popular, since it enables derivation of a closed form expression for the marginal likelihood under a conjugate prior. In our approach we bypass this restriction by replacing the true likelihood in the marginal likelihood integral by a pseudo-likelihood. This implies a factorization of the marginal pseudo-likelihood into terms that can be evaluated in closed form by using existing results for the marginal likelihoods of Gaussian directed acyclic graphs. The marginal pseudo-likelihood offers further advantages by allowing efficient search algorithms to be used, such that model optimization becomes realistic for even high-dimensional data.

Dobra et al. [19] considered a similar pseudo-likelihood based approach. These two methods involve similar techniques in the first step where a general dependency network is learned using a Bayesian approach. A dependency network [20] is a collection of conditional distributions for each variable given the others which are all fitted separately. In the general case, this network does not define a proper joint distribution for the variables. Dobra et al. use this dependency network in order to define an ordering for the variables before learning a directed acyclic graphical model over the variables. The found directed graph is then moralized in order to produce an undirected graph. In other words, their method does not consider general non-decomposable graphs.

Marginal pseudo-likelihood has been previously used to learn undirected graphical models with discrete variables in Pensar et al. [21]. Our paper can be seen to generalize the ideas developed there to the continuous domain by introducing the required methodology and providing a formal consistency proof under the multivariate normal assumption. Our method utilizes the fractional Bayes factors based approach of Consonni and La Rocca [22] to cope automatically with the difficulty of setting up prior distributions for the models' parameters.

The rest of the paper is organized as follows. After introducing the notation, we briefly review the results by Consonni and La Rocca that are needed in deriving the expression for the marginal pseudo-likelihood. In Section 3 we state our main result by introducing the fractional marginal pseudo-likelihood. The detailed proof of its consistency for Markov blanket estimation is given in Appendix. A score-based search algorithm adopted from Pensar et al. [21] is presented in order to implement the method in practice. In Section 4 we demonstrate the favourable performance of our method by several numerical experiments involving a comparison against `glasso`, `NBS` and `space`.

1.2. Notations and preliminaries

We will start by reviewing some of the basic concepts related to graphical models and the multivariate normal distribution. For a more comprehensive presentation, see for instance [2] and [3].

Consider an undirected graph $G = (V, E)$, where $V = \{1, \dots, p\}$ is the set of nodes (vertices) and $E \subset V \times V$ is the set of edges. There exists an (undirected) edge between the nodes i and j, if and only if $(i, j) \in E$ and $(j, i) \in E$. Each node of the graph corresponds to a random variable, and together they form a p-dimensional random vector $\boldsymbol{x}$. We will use the terms node and variable interchangeably. Absence of an edge in the graph G is a statement of conditional independence between the corresponding elements of $\boldsymbol{x}$. More in detail, $(i, j), (j, i) \notin E$ if and only if x_i and x_j are conditionally independent given the remaining variables $\boldsymbol{x}_{V\setminus\{i,j\}}$. This condition is usually referred as the pairwise Markov property. We let $mb(j)$ denote the Markov blanket of node j. The Markov blanket is defined as the set containing the neighbouring nodes of j, $mb(j) = \{i \in V \mid (i, j) \in E\}$. The local Markov property states that each variable is conditionally independent of all others given its Markov blanket. An undirected graph G is called decomposable or equivalently chordal if each cycle, whose length is greater or equal than 4, contains a chord. By a cycle, we mean a sequence of nodes such that the subsequent nodes are connected by an edge and the starting node equals the last node in the sequence. The length of a cycle equals the number of edges in the cycle. A chord is an edge between two non-subsequent nodes of the cycle.

We will write $\boldsymbol{x} \sim N_p(\boldsymbol{0}, \boldsymbol{\Omega}^{-1})$ to state that a random vector $\boldsymbol{x}$ follows a p-variate normal distribution with a zero mean and precision matrix $\boldsymbol{\Omega}$. We will denote the covariance matrix by $\boldsymbol{\Sigma} = \boldsymbol{\Omega}^{-1}$. The precision matrix $\boldsymbol{\Omega}$, and also equivalently $\boldsymbol{\Sigma}$, are always assumed to be symmetric and positive definite.

Given an undirected graph G and a random vector $\boldsymbol{x}$, we define a Gaussian graphical model to be the collection of multivariate normal distributions for $\boldsymbol{x}$ that satisfy the conditional independences implied by the graph G. Hence, a Gaussian graphical model consists of all the distributions $N_p(\boldsymbol{0}, \boldsymbol{\Omega}^{-1})$, where $\boldsymbol{\Omega}_{ij} = 0$ if and only if $(i, j) \notin E$, $i \neq j$. Otherwise, the elements of the inverse covariance matrix can be arbitrary, as long as symmetry and positive definiteness hold.

In contrast to the above undirected model, a Gaussian directed acyclic graphical model is a collection of multivariate normal distributions for $\boldsymbol{x}$, whose independence structure can be represented by some directed acyclic graph (DAG) $D = (V, E)$. When considering directed graphs, we use $pa(j)$ to denote the parent set of the node j. The set $pa(j)$ contains nodes i such that $(i, j) \in E$. That is, there exists a directed edge from i to j. Similar Markov assumptions as those characterizing the dependency structure under undirected models, as described above, hold also for directed models, see, for instance, [3]. For each decomposable undirected graph, we can find a DAG which defines the same conditional independence assertions. In general, the assertions representable by DAGs and undirected graphs are different.

2. Objective Comparison of Gaussian Directed Acyclic Graphs

Consonni and La Rocca [22] consider objective comparison of Gaussian directed acyclic graphical models and present a convenient expression for computing marginal likelihoods for any Gaussian DAG. Their approach to Gaussian DAG model comparison is based on using Bayes factors and uninformative, typically improper prior on the space of unconstrained covariance matrices. Ambiguity arising from the use of improper priors is dealt with by utilizing the fractional Bayes factors [23].

We first review a result concerning the computation of marginal likelihood in a more general setting, presented by Geiger and Heckerman [24]. They state five assumptions concerning the regularity of the sampling distribution of data and the structure of the prior distribution for parameters, that allow construction of parameter priors for every DAG model with a given set of nodes by specifying only one parameter prior for any of the complete DAG models. A complete DAG model refers to a model in which every pair of nodes is connected by an edge, implying that there are no conditional independence assertions between the variables. When the regularity assumptions are met, the following result can be derived:

Theorem 1. *(Theorem 2 in [24]) Let M and M_c be any DAG model and any complete DAG model for $\boldsymbol{x}$, respectively. Let $\boldsymbol{X}$ denote a complete (no missing observations) random sample of size n. Now the marginal likelihood for M is*

$$p(\boldsymbol{X} \mid M) = \prod_{j=1}^{p} \frac{p(\boldsymbol{X}_{fa(j)} \mid M_c)}{p(\boldsymbol{X}_{pa(j)} \mid M_c)}, \tag{1}$$

where $\boldsymbol{X}_{pa(j)}$ denotes the data belonging to the parents of x_j. We call $fa(j) = pa(j) \cup \{j\}$ the family of variable x_j.

Assumptions given by Geiger and Heckerman also imply that the marginal likelihood given by (1) scores all Markov equivalent DAGs equally, which is a desirable property when DAGs are considered only as models of conditional independence.

In order to apply (1), Consonni and La Rocca derive expressions for the marginal likelihoods corresponding to subvectors of $\boldsymbol{x}$, given the complete Gaussian DAG model. Objectivity is achieved by using an uninformative improper prior of the form

$$p(\boldsymbol{\Omega}) \propto |\boldsymbol{\Omega}|^{\frac{a_{\boldsymbol{\Omega}}-p-1}{2}}, \tag{2}$$

for the parameters of the complete DAG model. The improper prior is updated into a proper one by using fractional Bayes factors approach [23]. In this approach, a fraction of likelihood is "sacrificed" and used to update the improper prior into a proper fractional prior which is then paired with the remaining likelihood to compute the Bayes factors. Consonni and La Rocca show that the resulting fractional prior on the precision matrix $\boldsymbol{\Omega}$ is Wishart. This choice of prior combined with Gaussian likelihood satisfies all five assumptions required to use (1).

Setting $a_{\boldsymbol{\Omega}} = p - 1$ in (2), we can take the fraction of sacrificed likelihood to be $1/n$, see [22]. With this choice, the resulting fractional prior on $\boldsymbol{\Omega}$ is $W_p(p, (1/n)\boldsymbol{X}^T\boldsymbol{X})$, where $W_p(a, \boldsymbol{A})$ refers to a Wishart distribution (to provide interpretation for the parameters, a random variable following the distribution $W_p(a, A)$ would have an expected value of $a\boldsymbol{A}^{-1}$).

Now applying (1) and Eq. (25) in [22], we obtain the marginal likelihood of any Gaussian DAG as

$$p(\boldsymbol{X} \mid M) = \prod_{j=1}^{p} \pi^{-\frac{(n-1)}{2}} \frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{p_j+1}{2}\right)} n^{-\frac{2p_j+1}{2}} \left(\frac{|\boldsymbol{S}_{fa(j)}|}{|\boldsymbol{S}_{pa(j)}|}\right)^{-\frac{n-1}{2}}, \tag{3}$$

where p_j is the size of the set $pa(j)$, $\boldsymbol{S} = \boldsymbol{X}^T\boldsymbol{X}$ is the unscaled sample covariance matrix and $\boldsymbol{S}_A$ refers to a submatrix of $\boldsymbol{S}$ restricted to variables in the set A. The fractional marginal likelihood given by (3) is well defined if matrices $\boldsymbol{S}_{pa(j)}$ and $\boldsymbol{S}_{fa(j)}$ are positive definite for every j. This is satisfied with probability 1 if $n \geq \max\{p_j + 1 \mid j = 1, \ldots, p\}$.

Consonni and La Rocca also show that their methodology can be used to perform model selection among decomposable Gaussian graphical models. This is possible because every decomposable undirected graph is Markov equivalent to some DAG. A similar fractional marginal likelihood approach as presented above has been considered by Carvalho and Scott [6] but it was applied only in the context of decomposable Gaussian graphical models.

3. Structure Learning of Gaussian Graphical Models

3.1. Marginal Likelihood

Suppose we have a sample of independent and identically distributed multivariate normal data $\boldsymbol{X} = (\boldsymbol{X}^1, \ldots, \boldsymbol{X}^n)^T$, coming from a distribution whose conditional dependence structure is represented by an undirected graph G^*. We aim at identifying G^* based on $\boldsymbol{X}$, which is done with a Bayesian approach by maximizing the approximate posterior probability of the graph conditional on the data.

Posterior probability of a graph G given data $\boldsymbol{X}$ is proportional to

$$p(G \mid \boldsymbol{X}) \propto p(G)p(\boldsymbol{X} \mid G), \tag{4}$$

where $p(G)$ is the prior probability assigned to a specific graph and $p(\boldsymbol{X} \mid G)$ is the marginal likelihood. The normalizing constant of the posterior can be ignored, since it cancels in comparisons of different graphs. First, we focus on the marginal likelihood, since it is the data dependent term in (4). Later on, we will make use of the prior $p(G)$ term in order to promote sparsity in the graph structure.

By definition, the marginal likelihood of G equals

$$p(\boldsymbol{X} \mid G) = \int_{\Theta_G} p(\boldsymbol{\theta} \mid G)p(\boldsymbol{X} \mid \boldsymbol{\theta}, G)d\boldsymbol{\theta}, \tag{5}$$

where $\boldsymbol{\theta}$ is the parameter vector, $p(\boldsymbol{\theta} \mid G)$ denotes the parameter prior under G, the term $p(\boldsymbol{X} \mid \boldsymbol{\theta}, G)$ is the likelihood function and the integral is taken over the set of all possible parameters under G.

However, computing the marginal likelihood for a general undirected graph is very difficult, due the global normalizing constant in the likelihood term. Closed form solution exists only for chordal graphs, which is a highly restrictive assumption in general.

3.2. Marginal Pseudo-likelihood

We circumvent the problem of an intractable integration involved with the true likelihood function by using pseudo-likelihood. Pseudo-likelihood was introduced originally by Besag [25]. The idea behind the pseudo-likelihood can be motivated by thinking of it as an approximation for the true likelihood in form of a product of conditional probabilities or densities, where in each factor the considered variable is conditioned on all the rest. More formally, we write the pseudo-likelihood as

$$\hat{p}(\boldsymbol{X} \mid \boldsymbol{\theta}) = \prod_{j=1}^{p} p(X_j \mid \boldsymbol{X}_{-j}, \boldsymbol{\theta}),$$

where the notation $\boldsymbol{X}_{-j}$ stands for observed data on every variable except the j:th one.

In general, pseudo-likelihood should not be considered as a numerically exact and accurate approximation of the likelihood but as an object that has a computationally more attractive form and which can be used to obtain consistent estimates of parameters. It can be shown that under certain regularity assumptions, the pseudo-likelihood estimates for model parameters coincides with the maximum likelihood estimates, see [26].

One advantage of using pseudo-likelihood instead of the true likelihood is that it allows us to replace the global normalization constant by p local normalising constants related to conditional distributions of variables and thus makes the computations more tractable.

Using pseudo-likelihood, the original problem (5) of computing the marginal likelihood of $\boldsymbol{X}$ can be stated as

$$\begin{aligned} p(\boldsymbol{X} \mid G) &\approx \int_{\Theta_G} p(\boldsymbol{\theta} \mid G) \prod_{j=1}^{p} p(\boldsymbol{X}_j \mid \boldsymbol{X}_{-j}, \boldsymbol{\theta}, G) d\boldsymbol{\theta} \\ &= \hat{p}(\boldsymbol{X} \mid G) \end{aligned}$$

The term $\hat{p}(\boldsymbol{X} \mid G)$ is referred to as the marginal pseudo-likelihood, introduced by Pensar et al. [21] for discrete-valued undirected graphical models. The local Markov property states that given the variables in its Markov blanket $mb(j)$, the variable x_j is conditionally independent of the remaining variables. More formally, we have that

$$p(x_j \mid \boldsymbol{x}_{-j}, \boldsymbol{\theta}) = p(x_j \mid \boldsymbol{x}_{mb(j)}, \boldsymbol{\theta}).$$

Thus, we obtain the following form for the marginal pseudo-likelihood

$$\hat{p}(\boldsymbol{X} \mid G) = \int_{\Theta_G} p(\boldsymbol{\theta} \mid G) \prod_{j=1}^{p} p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)}, \boldsymbol{\theta}) d\boldsymbol{\theta} \tag{6}$$

We assume global parameter independence in order to factor the full integral into integrals over individual parameter sets Θ_j related to conditional distributions $p(x_j \mid \boldsymbol{x}_{mb(j)})$. The expression for the integral (6) becomes

$$\hat{p}(\boldsymbol{X} \mid G) = \prod_{j=1}^{p} \int_{\Theta_j} p(\boldsymbol{\theta}_j) p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)}, \boldsymbol{\theta}_j) d\boldsymbol{\theta}_j. \tag{7}$$

3.3. Fractional Marginal Pseudo-likelihood

The expression (7) for the marginal pseudo-likelihood can be regarded as a product of terms, where each term corresponds to a marginal likelihood of a DAG model. This offers a tractable way to compute the marginal pseudo-likelihood in closed form.

Recall the general formula for a marginal likelihood of any DAG model M, introduced in the previous section:

$$p(\boldsymbol{X} \mid M) = \prod_{j=1}^{p} \frac{p(\boldsymbol{X}_{fa(j)} \mid M_c)}{p(\boldsymbol{X}_{pa(j)} \mid M_c)} = \prod_{j=1}^{p} p(\boldsymbol{X}_j \mid \boldsymbol{X}_{pa(j)}, M_c), \tag{8}$$

where in the last equality we used the definition $fa(j) = \{j\} \cup pa(j)$.

We can see a clear resemblance between the forms (8) and (7). In both of these, each factor corresponds to a marginal likelihood of a DAG model, where we have a node and its parent nodes. In the case of Markov networks, the set of parents of a node is its Markov blanket, $mb(j)$.

Thus, we can use the closed form solution of (3) to compute the sought marginal pseudo-likelihood (7) by changing $pa(j) \rightarrow mb(j)$ and defining $fa(j) = \{j\} \cup mb(j)$. Then the closed form solution (3) for the fractional likelihood corresponds to

$$\begin{aligned} \hat{p}(\boldsymbol{X} \mid G) &= \prod_{j=1}^{p} \pi^{-\frac{(n-1)}{2}} \frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{p_j+1}{2}\right)} n^{-\frac{2p_j+1}{2}} \left(\frac{|\boldsymbol{S}_{fa(j)}|}{|\boldsymbol{S}_{mb(j)}|} \right)^{-\frac{n-1}{2}} \\ &= \prod_{j=1}^{p} p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)}), \end{aligned} \tag{9}$$

where $p_j = |mb(j)|$ and $\boldsymbol{S}$ refers to the full $p \times p$ unscaled sample covariance matrix. As before, $\boldsymbol{S}_{mb(j)}$ and $\boldsymbol{S}_{fa(j)}$ refer to submatrices of $\boldsymbol{S}$ restricted to variables in sets $mb(j)$ and $fa(j)$. From now on, $\hat{p}(\boldsymbol{X} \mid G)$ is referred to as fractional marginal pseudo-likelihood, due to the fractional Bayes factor approach used in derivation of the analytical form. The expression $p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)})$ in (9) is used to denote the local fractional marginal pseudo-likelihood for the node j.

The next theorem provides a theoretical justification for the approximation used in derivation of our scoring criterion.

Theorem 2. *Let* $\boldsymbol{x} \sim N_p(\mathbf{0}, (\boldsymbol{\Omega}^*)^{-1})$ *and* $G^* = (V, E^*)$ *denote the the undirected graph that completely determines the conditional independence statements between* $\boldsymbol{x}$*'s components. Let* $\{mb^*(1), \ldots, mb^*(p)\}$ *denote the set of Markov blankets, which uniquely define* G^**.*

Suppose we have a complete random sample $\boldsymbol{X}$ *of size n obtained from* $N_p(\mathbf{0}, (\boldsymbol{\Omega}^*)^{-1})$*. Then for every* $j \in V$*, the local fractional marginal pseudo-likelihood estimator*

$$\widehat{mb}(j) = \arg\max_{mb(j) \subset V \setminus \{j\}} p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)})$$

is consistent, that is, $\widehat{mb}(j) = mb^*(j)$ *with probability tending to* 1*, as* $n \to \infty$*.*

The detailed proof of Theorem 2 is presented in Appendix A. The proof is split in two parts; first, we show that the fractional marginal pseudo-likelihood score does not overestimate, *i.e.*, the true Markov blanket is preferred over the sets containing redundant nodes. The second part covers the underestimation: a set that does not contain all the members of the true Markov blanket will receive strictly lower score. Combining these two results implies our theorem. The strategy of dividing a proof in these kinds of cases is fairly common approach when proving the consistency of model selection criteria, see, for instance, [27] and [28]. The essential part in our proof is studying the asymptotic form of the data dependent term and showing that it behaves as desired in both of the required cases. The statements proven can be formulated into following lemmas:

Lemma 1. *Overestimation. Let* $mb^* \subset V \setminus \{j\}$ *and* $fa^* = mb^* \cup \{j\}$ *denote the true Markov blanket and the true family of the node* $j \in V$*, respectively. Let* $mb \subset V \setminus \{j\}$ *be a superset of the true Markov blanket,* $mb^* \subset mb$*. Now, as the sample size* $n \to \infty$

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb})} \to \infty$$

in probability.

Lemma 2. *Underestimation. Let* $mb^* \subset V \setminus \{j\}$ *and* $fa^* = mb^* \cup \{j\}$ *denote the true Markov blanket and the true family of the node* $j \in V$*, respectively. Assume that* $mb \subset mb^*$*. Let* $A \subset V \setminus fa^*$*. Now, as the sample size* $n \to \infty$

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^* \cup A})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb \cup A})} \to \infty$$

in probability.

In Lemma 2, we also allow for cases where $mb = \emptyset$ or $A = \emptyset$. With these proven, it is easy to see that our scoring function will asymptotically prefer the true Markov blanket over any other possible Markov blanket candidate. For supersets of the true Markov blanket, this follows from the overestimation lemma. For an arbitrary set that does not contain all the true members, we can apply the underestimation lemma to show that there is always a set with strictly higher score. This set is either the true Markov blanket or its superset. This suffices, since the latter case reduces to using the overestimation lemma again.

To be a bit more specific, consider a set mb which has the same cardinality as the true Markov blanket but does not contain all the true nodes. This set is not a superset, nor a subset of the true Markov blanket but it will receive a lower score asymptotically. This follows, since the underestimation lemma guarantees that a set that contains all the members of the true Markov blanket and the redundant ones from mb, will be preferred over mere mb. This reduces the problem to comparing the score of the true Markov blanket with its superset which is covered by the overestimation part.

The locally consistent Markov blankets imply that the whole graph is also asymptotically correctly estimated which is formulated in the following corollary:

Corollary 1. *Let* $\mathcal{G}$ *denote the set of all undirected graphs with p nodes. The global fractional marginal pseudo-likelihood estimator*

$$\widehat{G} = \arg\max_{G \in \mathcal{G}} \hat{p}(\boldsymbol{X} \mid G)$$

is consistent, that is, $\widehat{G} = G^*$ *with probability tending to* 1*, as* $n \to \infty$*.*

Proof. Theorem 2 guarantees that the true Markov blanket of each node is found with a probability tending to 1 as sample size increases. Since the structure of a Markov network is uniquely determined by its Markov blankets, the result follows. □

We will use the scoring function in conjunction with a search algorithm that finds the Markov blanket by incrementally adding and removing variables. To that end, we will need the following results in order to show the asymptotic correctness of the used algorithm with our score.

Lemma 3. *Assume that node i belongs to the Markov blanket of node j. Let $A \subset V \setminus \{j, i\}$ and denote $B = A \cup \{i\}$.*
Now, as the sample size $n \to \infty$

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_B)}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_A)} \to \infty$$

in probability.

The result above states that, asymptotically, adding a variable that is contained in the true Markov blanket will increase the score.

Lemma 4. *Let $mb^* \subset V \setminus \{j\}$ and $fa^* = mb^* \cup \{j\}$ denote true Markov blanket and the true family of the node $j \in V$, respectively. Let $R \subset V \setminus fa^*$, $R \neq \emptyset$ be some set of nodes not belonging to the true Markov blanket of j. Denote $A = mb^* \cup R$ and $B = mb^* \cup (R \setminus \{i\})$, where $i \in R$.*
Now, as the sample size $n \to \infty$

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_B)}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_A)} \to \infty$$

in probability.

Lemma 4 guarantees that removing a redundant node from a set that contains the true Markov blanket will increase the score. The statements of lemmas 3 and 4 are similar in spirit to the properties stated in Definition 6 by Chickering [29]. These properties were used by the author to prove the correctness of a greedy hill-climb algorithm operating in the space of DAGs.

3.4. Learning Algorithm for Markov Blanket Discovery

The consistency result of the local Markov blanket estimators stated in the last section allows us to optimize the Markov blanket of each variable independently which also makes this step extremely easy to compute in parallel. In practice, each search is done by implementing a greedy hill-climb algorithm similar to `interIAMB`-algorithm [30], with the heuristic function needed for measuring the association between two nodes given some others replaced by the fractional marginal pseudo-likelihood score. The same search procedure was used by Pensar et al. [21] for discrete Markov networks.

The search algorithm starts with an empty Markov blanket and then incrementally adds nodes that yield the maximum increase in the score. Each successful addition step is followed by a removal step, where nodes are removed from the blanket if this results in the increase of the score. The algorithm terminates and returns the estimated Markov blanket when there are no variables to be added so that the score would increase further. Pseudo-code of the algorithm is presented in Appendix C, Algorithm 1.

With help of the lemmas proved in the previous section, it is easy to see that the algorithm will asymptotically return the correct Markov blanket. More in detail, Lemma 3 implies that all the true variables will be included in the estimated blanket during the addition steps since these are guaranteed to increase the score. Redundant variables might also be included but as soon as we have found all the right ones, the Lemma 4 starts to apply, and we are guaranteed to remove the redundant ones in the removal step.

This reasoning follows a similar pattern as in [30] where the asymptotic correctness of `interIAMB` was shown under the assumption that the data generating distribution is faithful to some DAG. Our consistency proofs basically show that the fractional marginal pseudo-likelihood has the properties required by the sound heuristic function needed in the algorithm to measure dependency between variables.

Also, Peña et al. [31] proved that `IAMB`-algorithm, which differs slightly from `interIAMB` by not having the removal step after each successful addition but only in the end, is correct even when assumption about faithfulness

to some DAG is relaxed. One needs only to assume that the generating distribution has the composition property (see, Theorem 1 in [31]) and that the used function to measure dependence asymptotically decides on independence correctly.

In our implementation of the algorithm, we restricted the maximum possible Markov blanket size for each variable to be $n - 1$ when $n < p$. Otherwise the input empirical (unscaled) covariance matrix might not be positive definite which is required for our score to be defined. Also, the deletion step was conducted only if the current Markov blanket size was greater than two.

The proven consistency properties are asymptotic results so we are not guaranteed to produce proper undirected graphs on small sample sizes by naively combining each found Markov blanket together. To be more specific, we may find Markov blankets $mb(i)$ and $mb(j)$ such that $i \in mb(j)$ but $j \notin mb(j)$, which contradicts the definition of an undirected graph. To overcome this, we use two criteria, `AND` and `OR`, to combine the learned Markov blankets into proper undirected graphs.

Denote the identified Markov blankets by $mb(j),\ j = 1, \ldots, p$. The edge sets specifying `OR`- and `AND`-graphs are correspondingly defined as follows

$$E_{\texttt{OR}} = \{(i, j) \subset V \times V \mid i \in mb(j) \text{ or } j \in mb(i)\}$$
$$E_{\texttt{AND}} = \{(i, j) \subset V \times V \mid i \in mb(j) \text{ and } j \in mb(i)\}.$$

In addition to `AND`- and `OR`-method, we consider a third procedure referred to as the `HC`-method (Algorithm 2 in [21]). `HC`-method uses the graph obtained by `OR`-method to define a subspace of graphs $\mathcal{G}_{\texttt{OR}} = \{G = (V, E) \in \mathcal{G} |\ E \subset E_{\texttt{OR}}\}$. Then, starting from an empty graph, a simple deterministic greedy hill-climb based on local changes is performed in this reduced model space by removing or adding single edges resulting in the largest improvement in the fractional marginal pseudo-likelihood score. At each point, steps can be taken to neighbouring graphs, $N_{\mathcal{G}_{\texttt{OR}}}(G)$, which are graphs in $\mathcal{G}_{\texttt{OR}}$ that can be obtained from the current graph, G, by a single edge addition or deletion. Pseudo-code of the algorithm is presented in Appendix C.

Factorization of the fractional marginal pseudo-likelihood over the terms containing variables and their Markov blankets allows us compute the change in the score between two neighbouring graphs easily. For instance, when adding an undirected edge (i, j) we need only to compute four terms corresponding to the local scores for variables i and j given their new and old Markov blankets.

As mentioned in [21], the two-step strategy of first finding the possible Markov blankets and then running a score based hill-climb in reduced model space makes the `HC`-method similar in spirit to `Max-Min Hill-Climbing` -algorithm for learning DAGs by Tsamardinos et al. [32] with the difference being our proposed scoring function which is used in both steps of the algorithm.

3.5. Sparsity Promoting Prior Over Local Graphs

Until now we have assumed that every graph structure is *a priori* equally likely and thus the prior term $p(G)$ in (4) was ignored. However, in most applications with high-dimensional variable sets it is natural to assume that the underlying dependence structure is sparse. To promote sparsity beyond the basic Occam's razor, which is built into Bayesian model comparison, one can use the prior distribution $p(G)$ to penalize nodes for having too many elements in their Markov blankets. By defining our graph prior in terms of mutually independent prior beliefs about the Markov blankets, we maintain the useful factorization of our score and the local score is given by

$$p(mb(j))p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)}).$$

We start with a similar approach as used for example in Carvalho and Scott [6] to motivate our choice for the prior. In this approach, we assume that the inclusion of an edge in a graph happens with some unknown probability t, which corresponds to a successful Bernoulli trial. A finite sequence of these inclusions is a repeated Bernoulli trial and thus binomially distributed. We obtain the following form for the local prior

$$p(mb(j) \mid t) \propto t^{p_j}(1 - t)^{m - p_j}, \tag{10}$$

where p_j is the proposed size of the Markov blanket of j, or equivalently the number of edges connected to j (number of successes in repeated Bernoulli trials). We use m to represent the maximum number of edges, that could be present

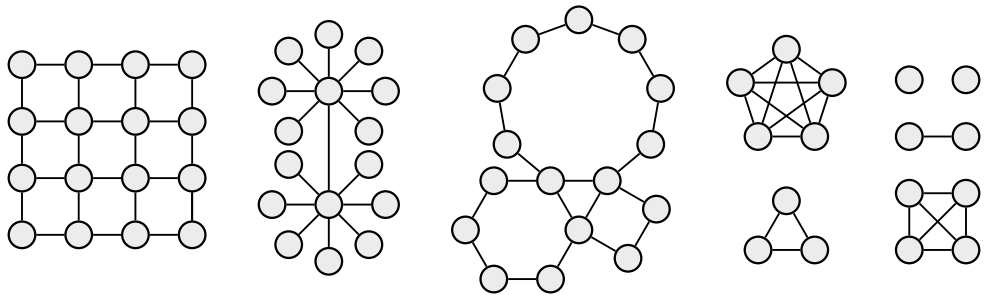

Figure 1: Synthetic subgraphs. Pictures appear originally in Pensar et al. [21].

in a local graph, that has $p_j + 1$ nodes. Hence m corresponds to the number of trials. Strictly speaking, such an interpretation is somewhat misleading since p_j can be at most $p - 1$ and $m = p_j(p_j + 1)/2$ depends on it. Carvalho and Scott [6] computed the prior probabilities for the whole graphs, not for the local graph structures as we here. So in their case $m = p(p-1)/2$ and the number of successful trials would simply equal the total amount of edges present in the graph.

Nevertheless, our slightly different approach defines a proper prior since the prior scores derived from equation (10) can be normalized by a constant that depends only on p, and thus cancels when comparing local graph structures. This prior is shown to perform favourably in the numerical tests considered later.

An appropriate value for the parameter t would be unknown for most applications. To overcome this issue, we put a prior on the parameter and integrate it out to obtain a suitable prior score function. Choosing a conjugate prior $t \sim \text{Beta}(a, b)$ and integrating leads to the expression

$$p(mb(j)) \propto \frac{\beta(a + p_j, b + m - p_j)}{\beta(a, b)},$$

where $\beta(\cdot, \cdot)$ refers to the beta function. In our numerical experiments, we use $a = b = 1/2$. Motivation for this choice is that $\text{Beta}(1/2, 1/2)$ is the Jeffreys' prior for the probability parameter of the binomial distribution, see, for instance, [33].

4. Numerical Experiments

4.1. Structure Learning with Synthetic Data

We first study the performance of the fractional marginal pseudo-likelihood in learning the graphical structures from synthetic multivariate normal data. We specify the structure of the generating network and measure the quality of the learned graphs using the Hamming distance which is defined as the number of edges to be added and deleted from a learned graph to obtain the true generating graph structure.

The synthetic graphs used to create our data are constructed by using 4 different subgraphs as building blocks. Graphs are shown in the Figure 1. Subgraphs are combined together as disconnected components to create a 64 node graph. This graph is again used as a component to build larger graphs. In total, the dimensions in the sequence $p = 64, 128, 256, 512, 1024$ are considered. The corresponding graphs contain $78, 156, 312, 624$ and 1248 edges, respectively.

When the graph structure is specified, we construct the corresponding precision matrices by setting elements to zeros as implied by the graph. The absolute values of the remaining off-diagonal elements are chosen randomly between 0.1 and 0.9 so that about half of the elements are negative. The diagonal elements are also first chosen randomly from the same interval and then a suitable vector is added to the diagonal in order to make all the eigenvalues positive, thus ascertaining the positive definiteness of the precision matrix. Finally, the matrix is inverted to get the covariance matrix and zero mean multivariate normal data is sampled using the built-in function 'mvnrnd' in Matlab.

For each of the considered dimensions, we created 25 covariance matrices, sampled a data set of 4000 observations and learned the structures using fractional marginal pseudo-likelihood, `glasso`, `space` and `NBS` with different sample

sizes. Input data were scaled so that each variable had zero mean and a standard deviation of one. The sparsity promoting prior was used with the fractional marginal pseudo-likelihood methods.

`Glasso` requires a user-specified tuning parameter that affects the sparsity of the estimated precision matrix. For every input data, we computed `glasso` using 12 different values for the tuning parameter logarithmically spaced on the interval $[0.01, 1]$. The best value for λ was chosen according to the extended BIC criterion proposed by Foygel and Drton [34]:

$$\texttt{EBIC}(\lambda) = n \operatorname{tr}(\hat{\boldsymbol{\Omega}}\boldsymbol{C}) - n \log\det(\hat{\boldsymbol{\Omega}}) + K \log n + 4K\gamma \log p,$$

where n denotes sample size, p is the number of variables, $\boldsymbol{C} = (1/n)\mathbf{S}$ is the maximum likelihood estimate for the covariance matrix, $\hat{\boldsymbol{\Omega}}$ stands for the estimate of the inverse covariance for given λ and K is the number of non-zero elements in the upper-half of $\hat{\boldsymbol{\Omega}}$, that is the number of edges in the corresponding graphical model. The parameter γ is constrained to be between 0 and 1. By using the value $\gamma = 0$, we would retain the ordinary BIC criterion, and increasing the γ would encourage sparser solutions. In the experiments, we used the value $\gamma = 0.5$.

The parameter value λ minimising the above criterion was used and the graphical model was read from the corresponding estimate of $\hat{\boldsymbol{\Omega}}$. R-package '`glasso`' [35] was used to perform the computations for `glasso`. The diagonal elements of the precision matrix were not penalized.

The computations for `NBS` were carried out using the Meinshausen-Bühlmann approximation also implemented in the R-package '`glasso`'. The required tuning parameter λ was chosen automatically, as proposed by the authors [17] to be $\lambda = (n^{-1/2})\Phi^{-1}(1 - \alpha/(2p^2))$, where $\alpha = 0.05$ and $\Phi(\cdot)$ denotes the c.d.f. of a standard normal random variable. Parameter α is related to the probability of falsely connecting two separate connectivity components of the true graph, see ch. 3 in Meinshausen and Bühlmann [17]. Since the resulting inverse covariance matrix, $\hat{\boldsymbol{\Omega}}$, was not necessarily symmetric, we used the average $(1/2)(\hat{\boldsymbol{\Omega}} + \hat{\boldsymbol{\Omega}}^T)$ to determine the estimated graph structure for `NBS`.

For the computations of `space` we used the corresponding R-package [36]. Also for this method, the user is required to specify a tuning parameter λ controlling the l_1-regularisation. We selected the scale of the tuning parameter to be $s = (n^{1/2})\Phi^{-1}(1 - \alpha/(2p^2))$ with $\alpha = 0.05$. Twelve candidate values for the tuning parameter were then chosen by multiplying a vector of 12 linearly space numbers from 0.5 to 3.25 by the scaling constant s. The best value for the λ was then chosen according to the BIC styled criterion proposed by the authors of the method (see ch. 2.4 in [18]). The `space` algorithm was run with uniform weights for regressions in the joint loss function and iteration parameter set to 2. For both `glasso` and `space` the range of possible tuning parameters was selected so that the best value according to the used criterion would lie strictly inside the given grid in all of the tests.

The Hamming distance results for the structure learning tests are shown in Figures 2 and 3. For the sake of clarity, `OR`- and `HC`-methods are omitted in Figure 2, and the comparison between fractional pseudo-likelihood is presented in Figure 3. The corresponding true positive and false positive rates for dimensions $d = 64$ and $d = 1024$ are presented in Table 1. All the shown results are averages computed from 25 data sets. The `AND`- and `HC`-method maintain almost equally good performance regardless the dimension considered and obtain the best overall performance in terms of Hamming distances. The `OR`-method is better on smaller dimensions where the graph is denser in the relative sense.

In the smaller dimensions `NBS` performs almost equally well as `AND` and `HC`. The graphs estimated by `NBS` are really sparse resulting in a low false positive rate. The Hamming distance curves of `glasso` do not seem to decrease consistently as the sample size grows. We tried also using the ordinary BIC-criterion for choosing the tuning parameter for `glasso` but this resulted in denser graphs and inferior Hamming distances (results not shown). The `space`-method improves its performance quite steadily as n grows and has nearly always the best true positive rate. However, this comes with a cost in terms of the false positive rate which is almost always higher for `space` than for the best pseudo-likelihood method or `NBS`. When the sample size is less than the dimension, `space` achieves good results with Hamming distances being equal or slightly better than those of `AND`-method. We can observe that in some settings, the results for `space` and `glasso` have relatively high standard deviations which demonstrates the sensitivity of these methods to the choice of tuning parameters with the used criteria.

To give a rough idea of the relative time complexity of the various methods, it took roughly half a second to estimate `OR`, `AND` and `HC` graphs in the $d = 64$ case when all the Markov blanket searches were run in a serial manner on a standard 2.3 GHz workstation. The high-dimensional cases were solved in couple minutes. Average running times of the other methods are tabulated in Appendix B. To summarize, the `NBS`-method was clearly the fastest, whereas `space` took the longest to run. `Space` was generally fast to compute when n was small but the running time varied considerably depending on the tuning parameter and grew quickly with the sample size. Even though computing a

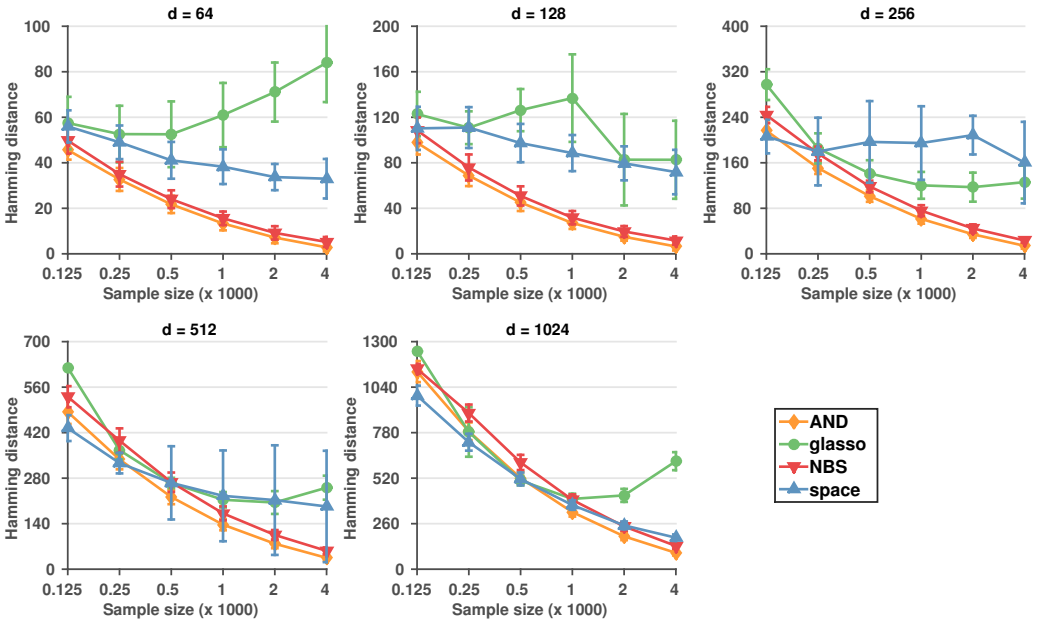

Figure 2: Sample size versus Hamming distance plots. Vertical lines show the standard deviations. Dimensions considered are $p = 64, 128, 256, 512$ and 1024.

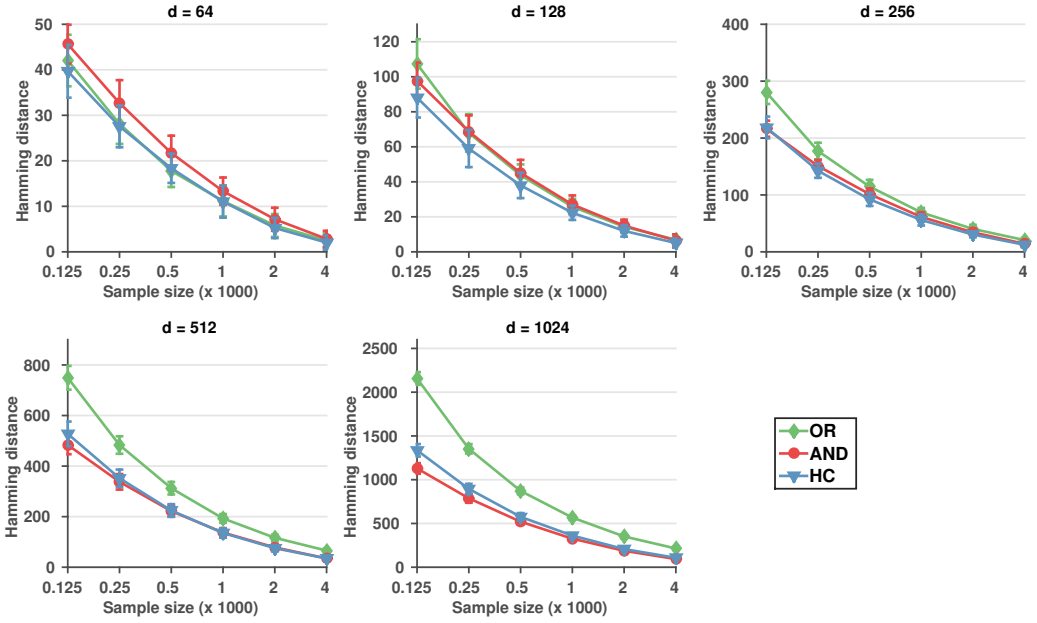

Figure 3: Sample size versus Hamming distance plots for `OR`, `AND` and `HC`. Vertical lines show the standard deviations.

p	n	OR		AND		HC		glasso		NBS		space	
		TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP
64	125	0.60	6e-03	0.44	9e-04	0.54	2e-03	0.53	1e-02	0.37	4e-04	0.66	2e-02
	250	0.72	3e-03	0.59	4e-04	0.68	1e-03	0.74	2e-02	0.57	6e-04	0.79	2e-02
	500	0.81	2e-03	0.73	2e-04	0.78	6e-04	0.86	2e-02	0.71	6e-04	0.88	2e-02
	1000	0.88	1e-03	0.83	1e-04	0.87	4e-04	0.93	3e-02	0.82	9e-04	0.94	2e-02
	2000	0.95	8e-04	0.91	6e-05	0.94	2e-04	0.97	4e-02	0.90	9e-04	0.98	2e-02
	4000	0.98	4e-04	0.96	4e-05	0.98	1e-04	0.99	4e-02	0.95	8e-04	0.99	2e-02
1024	125	0.43	3e-03	0.29	5e-04	0.37	1e-03	0.00	0	0.08	7e-07	0.36	4e-04
	250	0.61	2e-03	0.49	3e-04	0.57	7e-04	0.40	7e-05	0.29	1e-06	0.56	3e-04
	500	0.74	1e-03	0.66	2e-04	0.72	4e-04	0.66	2e-04	0.51	2e-06	0.72	3e-04
	1000	0.84	7e-04	0.79	1e-04	0.82	3e-04	0.81	3e-04	0.68	3e-06	0.83	3e-04
	2000	0.92	5e-04	0.88	7e-05	0.90	2e-04	0.91	6e-04	0.81	2e-06	0.91	3e-04
	4000	0.97	3e-04	0.94	5e-05	0.96	1e-04	0.96	1e-03	0.89	3e-06	0.96	3e-04

Table 1: A table showing true positive (TP) and false positive (FP) rates for different methods and sample sizes when $p = 64, 1024$. For the full table with all the dimensions, see Appendix B.

single instance of `glasso` or `space` might be faster than fractional pseudo-likelihood methods, one is usually forced to run these methods several times to find a suitable tuning parameter, thus making the actual running times much longer. Also, choosing an appropriate range for the candidate tuning parameters might prove difficult in some settings. These kind of practical difficulties make the method proposed here appealing, since no tuning parameters need to be chosen by the user.

Furthermore, running the Markov blanket searches in parallel provides an easy improvement in efficiency. To demonstrate the possible performance boost gained via parallellization, we also measured the times taken by each individual Markov blanket search. The average maximum times are tabulated in Appendix B. In the case with $d = 512$ and $n = 4000$, the longest time used for one Markov blanket search was on average 0.179 seconds. This represents roughly the time it would take for our method to estimate the graph if the computations were distributed among 512 cores.

4.2. Magnetic Resonance Data

We additionally illustrate the ability of the fractional marginal pseudo-likelihood to learn sparse structures by applying it to a real data set containing brain activity measurements. The whole data set consists of 2048 observations from a fMRI experiment on 90 variables corresponding to different regions of the brain. The data set is part of the R-package 'brainwaver' by Achard [37].

We used the first 50 variables and fitted a first-order vector autoregressive model to remove the most significant dependencies between subsequent sample vectors. As a result, we obtain 2048 residual vectors that should by assumption follow a multivariate normal distribution. The obtained data was then split into a training set and a test set. The size of the test set was always taken to be 48. For the training set size m, we considered three scenarios, where $m = 40$, $m = 200$ or $m = 2000$. Training data was always centered and scaled before applying methods. Centering of the test set was done using the means and standard deviations computed from the training data.

For pseudo-likelihood methods and `NBS` we first learned the graphical structure and then computed the maximum likelihood estimate for the precision matrix given the structure. In case of `glasso` and `space`, the precision matrix is readily available from the output of the algorithm. In these experiments we considered also the case where the sparsity promoting graph prior was not used with pseudo-likelihood methods.

For `glasso` we used 30 tuning parameters from the interval $[0.01, 10]$, choosing the best according to the extended BIC criterion. The `space`-method was also computed with 30 different tuning parameter values, scale selected as in the structure learning tests. Range of tuning parameters was again selected so that the best value according to the used BIC criterion would be strictly inside the grid. For `NBS` tuning parameter was chosen automatically as in the structure learning tests.

After the model learning, we took one data point at a time from the test set and tried to predict each of the components given the values of the others. Predicted value $\hat{X}_i$ for variable x_i was computed as $\hat{X}_i = \sum_{j \neq i} \rho_{ij} \sqrt{\omega_{jj}/\omega_{ii}} X_j$, where ω_{ii} are the diagonal elements of the estimated precision matrix and ρ_{ij} are the partial correlations which can be

m	OR	ORprior	glasso	NBS	space
40	1.002(11%)	**0.968**(4%)	1.080(0%)	1.057(0%)	0.988(4%)
200	**0.713**(11%)	0.722(7%)	0.923(3%)	0.721(7%)	0.717(16%)
2000	**0.647**(22%)	0.650(16%)	0.648(34%)	0.650(23%)	0.649(32%)

Table 2: A table showing average MSEs and edge densities (in parentheses) for different methods applied to brain data residuals.

obtained from the precision matrix. Squared difference of predicted value to the real value was recorded and the mean squared error (MSE) was used to compare the predictive performances of different methods.

Table 2 shows the results of prediction tests for training sample sizes $m = 40$, $m = 200$ and $m = 2000$. Results for AND and HC methods are omitted, since these were generally slightly worse than the results of OR. Here, we use OR to denote the method without the sparsity promoting prior whereas ORprior refers to the one with it. The shown results are averages from 50 tests. We can observe that OR with a graph prior provides the lowest prediction error when the sample size is less than the dimension. When the number of observations grows, OR without prior obtains the best predictions. In general, the differences between the methods are quite marginal. However, the models estimated by OR are usually substantially sparser than the ones estimated by competing methods, especially with the highest sample size considered here, $m = 2000$. Sparse models are naturally desirable as they are easier to interpret. In addition to that, these conditional independences captured by OR are relevant in a sense that the corresponding model provides the lowest MSEs when predicting missing data.

4.3. Gene Expression Data

We conducted similar prediction experiments using gene expression data from [38]. The original data set contains 335 microarray observations on 48701 variables. We formed three smaller sets of this data by first conducting a Kolmogorov-Smirnov test for each variable and then grouping them using the p-value obtained from the test. In the Kolmogorov-Smirnov test, the null-hypothesis is that data comes from a normal distribution. The aim of this procedure was to separate variables roughly according to how Gaussian their marginal distributions look like, and see if this has effect on the performance in prediction.

Data set 1 consisted of variables for which the obtained p-value was greater than 0.9, data set 2 were variables with p-value between 0.4 and 0.6, and data set 3 included the variables with the p-value less than 0.01. Finally, we randomly picked 100 variables from each of these three sets to the actual tests.

For each of the sets, we split the observations into a training set of size 300 and test set containing the rest 35 observations. The procedure described with the brain data was then repeated individually for each set of variables. In these experiments, we used 15 tuning parameters for glasso and space. The results are shown in Figure 4.

We can observe that all the methods perform almost equally well on the first two sets which represent the most Gaussian variables. The OR-method, which is the best out of fractional marginal pseudo-likelihood methods, has slightly higher MSE but this is obtained with substantially sparser model compared to other approaches. In the final set, MSE is generally higher for all the methods and differences between the methods are clearer. Here, OR achieves the best MSE with a marginal difference to space. The model is sparser than the one produced by space but more dense than those of glasso or NBS.

4.4. Flow Cytometry Data

We analyzed a flow cytometry data set found in R-package 'FBFsearch' [39]. This data is originally from [40] where the authors infer a causal Bayesian network depicting the signaling pathway between the proteins. The resulting network is reported to highly agree with the previous findings on the known relationships between proteins. The data set consists of 7466 observations on 11 variables. The number of variables is quite small but it nevertheless provides an interesting target for the structure learning methods since we have a "ground truth" to compare against.

We normalized the data by centering and scaling variables to have a standard deviation of one. The fractional pseudo-likelihood methods were run both with the sparsity promoting prior and without it. The tuning parameters for the other methods were selected using the same criteria as before. We used 30 candidate values for glasso and space. The quality of the learned network was measured with Hamming distance. The true directed graph was converted to

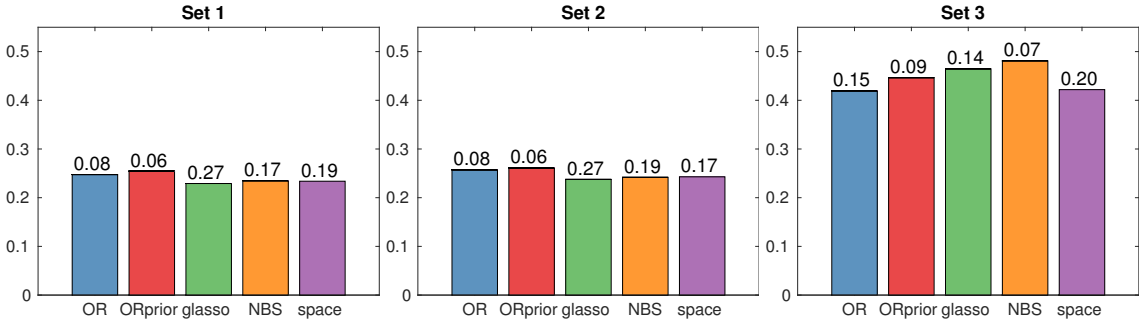

Figure 4: Mean squared errors for different methods and different sets of variables. Heights of the bars represent MSEs and the density of the corresponding graphical model is given on top of the bar. The shown values are averages computed from 10 tests (10 different partitions to a training and a test set).

an undirected one by simply omitting the edge orientations. One could also use the moral graph corresponding to the directed graph as the ground truth graph. The true graph would then have two extra edges that are not present in the undirected skeleton. However, these two edges are not found by the methods shown in the results and it would not affect the order of how the methods performed.

The resulting graphs for the best methods, `AND` and `NBS`, are presented in Figure 5. The `AND`-method without the prior achieved the lowest Hamming distance of 16. The performance of the `NBS`-method was similar, resulting in a Hamming distance of 18. This is also the same Hamming distance that was achieved by the `AND`-method with prior. `OR` and `HC` methods achieve slightly worse Hamming distances, ranging from 20 to 23. The used criteria for tuning parameter selection seemed to favour nearly fully connected networks for `space` and `glasso` which also resulted in substantially higher Hamming distances (results not shown).

However, among the tested tuning parameter values for `space` and `glasso`, one could find a value that resulted in a Hamming distance of 15 but it was not selected by the used criteria. This provides yet another evidence on the difficulty of choosing an appropriate tuning parameter for structure learning and lends support to our proposed approach where the selection is handled automatically.

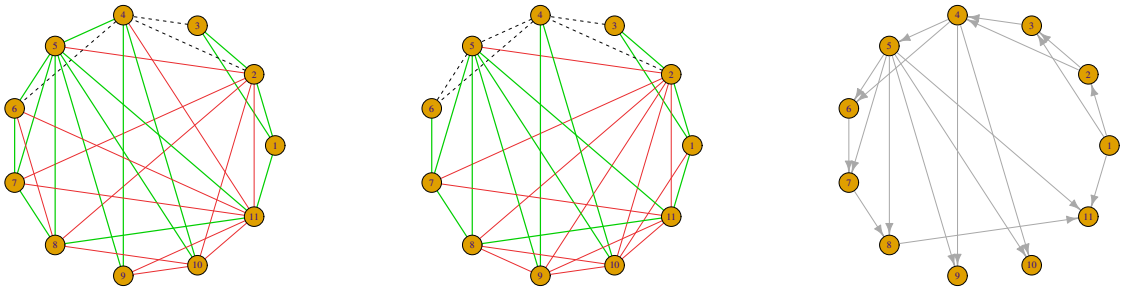

Figure 5: **Left:** A graph estimated by `AND`-method. **Middle:** A graph estimated by `NBS` approach. **Right:** A directed acyclic graph depicting the known regulatory network. In the estimated graphs, red edges represent false positives, green are true positives and dashed lines depict false negatives.

In addition to the prediction tests with the real data, we conducted similar tests using data from the same synthetic networks as in the structure learning tests. In these tests, fractional marginal pseudo-likelihood based methods achieved slightly better predictions. These results are presented in detail in Appendix B.

To conclude, we emphasize that the results produced by our methods are achieved without needing to tune any hyperparameters. Only choice left to the user is whether to include the sparsity promoting prior or not. This can be

a drastic advantage as demonstrated by the structure learning tests, where the suggested criteria for tuning parameter selection did not seem to be always optimal if the goal is to find the correct graphical structure. However, if the goal is to select a model with a good prediction power, the differences were not substantial and the used criteria produced good results.

5. Discussion

In this work we have introduced the fractional marginal pseudo-likelihood, an approximate Bayesian method for learning graphical models from multivariate normal data. One particular advantage of the method is its objectivity, since it does not necessitate the use of any domain specific knowledge which may be difficult to elicitate and use in general. That is, the method does not require the user to provide any hyperparameters that would affect the result, only choice is to decide whether to use sparsity promoting prior or not.

The method outputs three graphs corresponding to `AND`-, `OR`- and `HC`-procedures. The two first mentioned are obtained with the same computational effort. Based on our experiments, each graph has a clear interpretation in terms of when one should use it. If the goal is to use the model for prediction, `OR` is the best choice. In case one is interested in an easily interpretable graph and false positive edges are unwanted, we suggest choosing the `AND`-method. In a general case, one should stick to using `HC`, as this provides a compromise between these two alternatives.

In addition, the method allows graphs to be non-decomposable, which can be of substantial importance in applications. Earlier research has demonstrated that when the data generating process deviates from decomposability, graph learning methods building on the assumption of decomposability tend to yield unnecessarily dense graphs resulting from addition of spurious edges to chordless cycles.

As shown formally, our method enjoys consistency and was found in simulation experiments to yield essentially at least as accurate estimates of the graph structure as the competing methods without needing to tune any hyperparameters. For many applications of graphical model learning it is essential to retain solid interpretability of the estimated covariance structure, which means that high fidelity of the graph structure estimator is the most desirable property. In particular, frequently arising spurious edges may lead to confusing interpretations in the high-dimensional setting. In terms of predictive performance, methods considered here delivered similar levels of accuracy. Our divide-and-conquer type solution offers a possibility for efficient parallelization, as the initial Markov blanket search can be performed independently for each node. Hence, an attractive target for future research would include applications to very high-dimensional data sets and development of various parallelization schemes. In addition, it would be interesting to investigate altered versions by making the method more robust to outliers through relaxing the Gaussian assumption. The robust method could for instance be compared against a method by Sun and Li [41], which was shown to perform better than `glasso` when the data follow a heavier tailed distribution than a Gaussian.

Acknowledgements

The research in this article was financially supported by the COIN Centre of Excellence. We would like to thank the two anonymous referees whose comments helped us to improve the paper considerably.

References

[1] A. P. Dempster, Covariance Selection, Biometrics 28 (1) (1972) 157–175.
[2] J. Whittaker, Graphical Models in Applied Multivariate Statistics, John Wiley & Sons, 1990.
[3] S. Lauritzen, Graphical Models, Clarendon Press, 1996.
[4] B. Jones, C. Carvalho, A. Dobra, C. Hans, C. Carter, M. West, Experiments in Stochastic Computation for High-Dimensional Graphical Models, Statistical Science 20 (4) (2005) 388–400.
[5] J. G. Scott, C. M. Carvalho, Feature-Inclusion Stochastic Search for Gaussian Graphical Models, Journal of Computational and Graphical Statistics 17 (4) (2008) 790–808.
[6] C. M. Carvalho, J. G. Scott, Objective Bayesian model selection in Gaussian graphical models, Biometrika 96 (3) (2009) 497–512.
[7] A. M. Fitch, M. B. Jones, H. Massam, The Performance of Covariance Selection Methods That Consider Decomposable Models Only, Bayesian Analysis 9 (3) (2014) 659–684.
[8] F. Wong, C. K. Carter, R. Kohn, Efficient estimation of covariance selection models, Biometrika 90 (4) (2003) 809–830.
[9] A. Atay-Kayis, H. Massam, A Monte Carlo method for computing the marginal likelihood in nondecomposable Gaussian graphical models, Biometrika 92 (2) (2005) 317–335.

[10] B. Moghaddam, E. Khan, K. P. Murphy, B. M. Marlin, Accelerating Bayesian Structural Inference for Non-Decomposable Gaussian Graphical Models, in: Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, A. Culotta (Eds.), Advances in Neural Information Processing Systems 22, Curran Associates, Inc., 1285–1293, 2009.
[11] A. Dobra, A. Lenkoski, A. Rodriguez, Bayesian Inference for General Gaussian Graphical Models With Application to Multivariate Lattice Data, Journal of the American Statistical Association 106 (496) (2011) 1418–1433.
[12] A. Mohammadi, E. C. Wit, Bayesian Structure Learning in Sparse Gaussian Graphical Models, Bayesian Analysis 10 (1) (2015) 109–138.
[13] F. Stingo, G. M. Marchetti, Efficient local updates for undirected graphical models, Statistics and Computing 25 (1) (2015) 159–171.
[14] H. Wang, Scaling It Up: Stochastic Search Structure Learning in Graphical Models, Bayesian Analysis 10 (2) (2015) 351–377.
[15] J. Friedman, T. Hastie, R. Tibshirani, Sparse inverse covariance estimation with the graphical lasso, Biostatistics 9 (3) (2008) 432–441.
[16] D. M. Witten, J. H. Friedman, N. Simon, New Insights and Faster Computations for the Graphical Lasso, Journal of Computational and Graphical Statistics 20 (4) (2011) 892–900.
[17] N. Meinshausen, P. Bühlmann, High-dimensional graphs and variable selection with the Lasso, The Annals of Statistics 34 (3) (2006) 1436–1462.
[18] J. Peng, P. Wang, N. Zhou, J. Zhu, Partial Correlation Estimation by Joint Sparse Regression Models, Journal of the American Statistical Association 104 (486) (2009) 735–746.
[19] A. Dobra, C. Hans, B. Jones, J. R. Nevins, G. Yao, M. West, Sparse graphical models for exploring gene expression data, Journal of Multivariate Analysis 90 (1) (2004) 196–212, special Issue on Multivariate Methods in Genomic Data Analysis.
[20] D. Heckerman, D. M. Chickering, C. Meek, R. Rounthwaite, C. Kadie, Dependency Networks for Inference, Collaborative Filtering, and Data Visualization, Journal of Machine Learning Research 1 (2001) 49–75.
[21] J. Pensar, H. Nyman, J. Niiranen, J. Corander, Marginal pseudo-likelihood learning of discrete Markov network structures, Bayesian Analysis Advance Publication (2016) DOI: 10.1214/16-BA1032.
[22] G. Consonni, L. La Rocca, Objective Bayes Factors for Gaussian Directed Acyclic Graphical Models, Scandinavian Journal of Statistics 39 (4) (2012) 743–756.
[23] A. O'Hagan, Fractional Bayes Factors for Model Comparison, Journal of the Royal Statistical Society. Series B (Methodological) 57 (1) (1995) 99–138.
[24] D. Geiger, D. Heckerman, Parameter priors for directed acyclic graphical models and the characterization of several probability distributions, The Annals of Statistics 30 (5) (2002) 1412–1440.
[25] J. E. Besag, Nearest-Neighbour Systems and the Auto-Logistic Model for Binary Data, Journal of the Royal Statistical Society. Series B (Methodological) 34 (1) (1972) 75–83.
[26] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.
[27] D. M. A. Haughton, On the Choice of a Model to Fit Data from an Exponential Family, The Annals of Statistics 16 (1) (1988) 342–355.
[28] C. Z. Wei, On Predictive Least Squares Principles, The Annals of Statistics 20 (1) (1992) 1–42.
[29] D. M. Chickering, Optimal Structure Identification with Greedy Search, Journal of Machine Learning Research 3 (2003) 507–554.
[30] I. Tsamardinos, C. F. Aliferis, A. Statnikov, Algorithms for Large Scale Markov Blanket Discovery, in: I. Russell, S. Haller (Eds.), The 16th International FLAIRS Conference, AAAI Press, 376–380, 2003.
[31] J. M. Peña, R. Nilsson, J. Bjrkegren, J. Tegnr, Towards scalable and data efficient learning of Markov boundaries, International Journal of Approximate Reasoning 45 (2) (2007) 211 – 232.
[32] I. Tsamardinos, L. E. Brown, C. F. Aliferis, The max-min hill-climbing Bayesian network structure learning algorithm, Machine Learning 65 (1) (2006) 31–78.
[33] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, D. B. Rubin, Bayesian Data Analysis, Chapman and Hall/CRC, 2014.
[34] R. Foygel, M. Drton, Extended Bayesian Information Criteria for Gaussian Graphical Models, in: J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, A. Culotta (Eds.), Advances in Neural Information Processing Systems 23, Curran Associates, Inc., 604–612, 2010.
[35] J. Friedman, T. Hastie, R. Tibshirani, R-package glasso: Graphical lasso- estimation of Gaussian graphical models, version 1.8, http://cran.r-project.org/web/packages/glasso/, 2014 (accessed 27.9.16).
[36] J. Peng, P. Wang, N. Zhou, J. Zhu, R-package space: Sparse PArtial Correlation Estimation, version 0.1-1, https://cran.r-project.org/web/packages/space/, 2010 (accessed 27.9.16).
[37] S. Achard, R-package brainwaver: Basic wavelet analysis of multivariate time series with a visualisation and parametrisation using graph theory, version 1.6, http://cran.r-project.org/web/packages/brainwaver/, 2012 (accessed 27.9.16).
[38] J. Hiissa, L. L. Elo, K. Huhtinen, A. Perheentupa, M. Poutanen, T. Aittokallio, Resampling reveals sample-level differential expression in clinical genome-wide studies, OMICS: A Journal of Integrative Biology 13 (5) (2009) 381–396.
[39] D. Altomare, G. Consonni, L. La Rocca, R-package FBFsearch: Algorithm for searching the space of Gaussian directed acyclic graphical models through moment fractional Bayes factors, version 1.0, https://cran.r-project.org/web/packages/FBFsearch/, 2013 (accessed 27.9.16).
[40] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, G. P. Nolan, Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data, Science 308 (5721) (2005) 523–529.
[41] H. Sun, H. Li, Robust Gaussian Graphical Modeling Via l1 Penalization, Biometrics 68 (4) (2012) 1197–1206.
[42] J. S. Press, Applied Multivariate Analysis: Using Bayesian and Frequentist Method of Inference, Robert E. Krieger Publishing Company, 1982.
[43] P. Baldi, S. Brunak, Y. Chauvin, C. A. F. Andersen, H. Nielsen, Assessing the accuracy of prediction algorithms for classification: an overview, Bioinformatics 16 (5) (2000) 412–424.

Appendix A. Consistency Proofs

This section contains the proofs of Lemmas 1 and 2 which together imply the consistency of our method as formulated in Theorem 2 and Corollary 1. We follow the same notation and the assumptions given in Theorem 2. We

prove also Lemmas 3 and 4 which show the correctness of the used search algorithm.

The following proposition found in [2] is used in the proof.

Theorem 3. *(Based on* 6.7.1; *p.* 179) *Suppose the normal random vector* $\boldsymbol{x}$ *can be partitioned into three* $(\boldsymbol{x}_A, \boldsymbol{x}_B, \boldsymbol{x}_C)$ *and all conditional independence constraints can be summarised by the single statement* $\boldsymbol{x}_B \perp\!\!\!\perp \boldsymbol{x}_C \mid \boldsymbol{x}_A$. *If* $\boldsymbol{x}_A, \boldsymbol{x}_B$ *and* $\boldsymbol{x}_C$ *are p-,q- and r-dimensional respectively, then the deviance*

$$\text{dev}(\boldsymbol{x}_B \perp\!\!\!\perp \boldsymbol{x}_C \mid \boldsymbol{x}_A) = -n \log \frac{|\boldsymbol{S}||\boldsymbol{S}_A|}{|\boldsymbol{S}_{A\cup B}||\boldsymbol{S}_{A\cup C}|}$$

has an asymptotic chi-squared distribution with qr degrees of freedom.

Here $\boldsymbol{S}$ is defined as before, but in [2], $\boldsymbol{S}$ is used to denote the sample covariance matrix. It is clear that this does not change the statement of the theorem in any manner of consequence to our purposes. Note that theorem holds also if $A = \emptyset$, since complete independence can be considered a special case of the conditional independence. In this case, term $|\boldsymbol{S}_A|$ in the expression of deviance simply disappears.

Appendix A.1. Overestimation (Lemma 1)

Let $mb^* \subset V \setminus \{j\}$ and $fa^* = mb^* \cup \{j\}$ denote the true Markov blanket and the true family of the node x_j, respectively. We denote the cardinality of mb^* by p_j. Let $mb \subset V \setminus \{j\}$ be a superset of the true Markov blanket mb^*. Denote $a = |mb| - p_j$. Since $mb^* \subset mb$, we have $a > 0$.

We want to show that

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb})} \to \infty$$

in probability, as $n \to \infty$. Showing this will guarantee that fractional marginal pseudo-likelihood prefers the true Markov blanket over its supersets as the sample size increases. Remember, that the local fractional marginal pseudo-likelihood for $mb(j)$ was given according to

$$p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)}) = \pi^{-\frac{(n-1)}{2}} \frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{p_j+1}{2}\right)} n^{-\frac{2p_j+1}{2}} \left(\frac{|\boldsymbol{S}_{fa(j)}|}{|\boldsymbol{S}_{mb(j)}|}\right)^{-\frac{n-1}{2}}.$$

Consider next the log ratio of local fractional marginal pseudo-likelihoods, for mb^* and mb. The term containing the power of π appears in both of the terms, and so it cancels. By noticing that

$$n^{-\left(\frac{1+2p_j}{2}\right)} \Big/ n^{-\left(\frac{1+2(p_j+a)}{2}\right)} = n^a,$$

we get the following form for the ratio

$$\begin{aligned}\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb})} &= \log \frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{n+p_j+a}{2}\right)} + \log \frac{\Gamma\left(\frac{1+p_j+a}{2}\right)}{\Gamma\left(\frac{1+p_j}{2}\right)} \\ &\quad + a \log n - \left(\frac{n-1}{2}\right) \log \left(\frac{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}{|\boldsymbol{S}_{mb^*}||\boldsymbol{S}_{fa}|}\right). \end{aligned} \tag{A.1}$$

The second term in (A.1) doesn't depend on n so it can be omitted when considering the leading terms as $n \to \infty$. Denote $m = (n + p_j)/2$. Clearly $m \to \infty$, as $n \to \infty$. Now we can write the first term in (A.1) as

$$\log \frac{\Gamma(m)}{\Gamma\left(m + \frac{a}{2}\right)} = \log \Gamma(m) - \log \Gamma\left(m + \frac{a}{2}\right). \tag{A.2}$$

Now letting $n \to \infty$ and by using Stirling's asymptotic formula for each of the terms in (A.2), we get

$$\begin{aligned}\log\Gamma(m) - \log\Gamma\left(m + \frac{a}{2}\right) &= \left(m - \frac{1}{2}\right)\log m - m \\ &- \left(\left(m + \frac{a}{2} - \frac{1}{2}\right)\log\left(m + \frac{a}{2}\right) - \left(m + \frac{a}{2}\right)\right) + O(1).\end{aligned}$$

We see that m-terms cancel and the constant $a/2$ in the second term can be omitted. After rearranging the terms, the result can be written as

$$m\log\left(\frac{m}{m+\frac{a}{2}}\right) + \frac{1}{2}\log\left(\frac{m+\frac{a}{2}}{m}\right) - \frac{a}{2}\log\left(m + \frac{a}{2}\right) + O(1).$$

As $n \to \infty$, we have that

$$m\log\left(\frac{m}{m+\frac{a}{2}}\right) = \frac{1}{2}\log\left(\frac{1}{1+\frac{a}{2m}}\right)^{2m} \to \frac{1}{2}\log(\exp(-a)) = -\frac{a}{2}$$

and

$$\frac{1}{2}\log\left(\frac{m+\frac{a}{2}}{m}\right) = \frac{1}{2}\log\left(1 + \frac{a}{2m}\right) \to 0.$$

Thus, we can write (A.2) asymptotically as

$$\log\frac{\Gamma(m)}{\Gamma\left(m+\frac{a}{2}\right)} = -\frac{a}{2}\log\left(m + \frac{a}{2}\right) + O(1),$$

or equivalently by using variable n

$$\log\frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{n+p_j+a}{2}\right)} = -\frac{a}{2}\log\left(\frac{n+p_j+a}{2}\right) + O(1).$$

Now we can simplify the original formula (A.1) by combining the first and the third term

$$\begin{aligned}\log\frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{n+p_j+a}{2}\right)} + a\log n &= -\frac{a}{2}\log\left(\frac{n+p_j+a}{2}\right) + \frac{a}{2}\log n^2 + O(1) \\ &= \frac{a}{2}\log n + O(1).\end{aligned}$$

Consider next the last term in (A.1)

$$-\left(\frac{n-1}{2}\right)\log\left(\frac{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}{|\boldsymbol{S}_{mb^*}||\boldsymbol{S}_{fa}|}\right). \tag{A.3}$$

Since $mb^* \subset mb$, we can write $mb = mb^* \cup R$, where R denotes the set of redundant variables in mb. Recall the Theorem 3 and notice that by denoting

$$A = mb^*,\ B = \{j\} \text{ and } C = R,$$

it holds that $\boldsymbol{x}_B \perp\!\!\!\perp \boldsymbol{x}_C \mid \boldsymbol{x}_A$, since mb^* was the true Markov blanket of x_j. Note also that in this case $qr = 1 \cdot a = a$. Now the deviance can be written as

$$\text{dev}(x_j \perp\!\!\!\perp \boldsymbol{x}_R \mid \boldsymbol{x}_{mb^*}) = -n\log\left(\frac{|\boldsymbol{S}_{fa}||\boldsymbol{S}_{mb^*}|}{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}\right),$$

which is essentially just the determinant term (A.3) multiplied by a constant -2. Let us denote $D_n = \text{dev}(x_j \perp\!\!\!\perp \boldsymbol{x}_R \mid \boldsymbol{x}_{mb^*})$. The determinant term gets the following representation

$$\begin{aligned}-\left(\frac{n-1}{2}\right)\log\left(\frac{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}{|\boldsymbol{S}_{mb^*}||\boldsymbol{S}_{fa}|}\right) &= -\frac{n}{2}\log\left(\frac{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}{|\boldsymbol{S}_{mb^*}||\boldsymbol{S}_{fa}|}\right) + O_p(1) \\ &= -\frac{D_n}{2} + O_p(1).\end{aligned}$$

The $O_p(1)$ error on the first line comes from omitting the term

$$\frac{1}{2}\log\left(\frac{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}{|\boldsymbol{S}_{mb^*}||\boldsymbol{S}_{fa}|}\right).$$

Asymptotically, it holds that $D_n \sim \chi^2_a$. In other words the sequence (D_n) converges in distribution to a random variable D, where $D \sim \chi^2_a$. Convergence in distribution implies that the sequence (D_n) is bounded in probability, that is, $D_n = O_p(1)$ for all n.

Combining the above findings, asymptotically

$$-\left(\frac{n-1}{2}\right)\log\left(\frac{|\boldsymbol{S}_{fa^*}||\boldsymbol{S}_{mb}|}{|\boldsymbol{S}_{mb^*}||\boldsymbol{S}_{fa}|}\right) = O_p(1).$$

Adding the results together, we have shown that, as $n \to \infty$

$$\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb})} = \frac{a}{2}\log n + O_p(1).$$

Now since $a > 0$, then

$$\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb})} \to \infty$$

in probability, as $n \to \infty$.

Appendix A.2. Underestimation (Lemma 2)

Let mb^* denote the true Markov blanket of node x_j and $mb \subset mb^*$. Let $A \subset V \backslash fa^*$. Remember that fa^* was defined to be $mb^* \cup \{j\}$. Note that A could also be an empty set. We want to show that

$$\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*\cup A})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb\cup A})} \to \infty$$

in probability, as $n \to \infty$. Denote $|mb^* \cup A| = p_j$ and $a = |mb \cup A| - p_j$. Here $a < 0$, since mb is a subset of the true Markov blanket. We can now proceed similarly as in the overestimation part, and write the log ratio as

$$\begin{aligned}\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*\cup A})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb\cup A})} &= \log\frac{\Gamma\left(\frac{n+p_j}{2}\right)}{\Gamma\left(\frac{n+p_j+a}{2}\right)} + \log\frac{\Gamma\left(\frac{1+p_j+a}{2}\right)}{\Gamma\left(\frac{1+p_j}{2}\right)} \\ &\quad + a\log n - \left(\frac{n-1}{2}\right)\log\left(\frac{|\boldsymbol{S}_{fa^*\cup A}||\boldsymbol{S}_{mb\cup A}|}{|\boldsymbol{S}_{mb^*\cup A}||\boldsymbol{S}_{fa\cup A}|}\right). \qquad \text{(A.4)}\end{aligned}$$

The first three terms are just the same ones appearing in (A.1), which allows us to write

$$\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*\cup A})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb\cup A})} = \frac{a}{2}\log n - \left(\frac{n-1}{2}\right)\log\left(\frac{|\boldsymbol{S}_{fa^*\cup A}||\boldsymbol{S}_{mb\cup A}|}{|\boldsymbol{S}_{mb^*\cup A}||\boldsymbol{S}_{fa\cup A}|}\right) + O(1). \qquad \text{(A.5)}$$

Consider next the determinant term in (A.5)

$$-\left(\frac{n-1}{2}\right)\log\left(\frac{|\boldsymbol{S}_{fa^*\cup A}||\boldsymbol{S}_{mb\cup A}|}{|\boldsymbol{S}_{mb^*\cup A}||\boldsymbol{S}_{fa\cup A}|}\right). \qquad \text{(A.6)}$$

By the definition of $\boldsymbol{S}$, it is clear that

$$\frac{\boldsymbol{S}}{n} = \frac{1}{n}\boldsymbol{X}^T\boldsymbol{X} = \hat{\boldsymbol{\Sigma}},$$

where $\hat{\boldsymbol{\Sigma}}$ is the maximum likelihood estimate of the true covariance matrix. As n approaches infinity, the maximum likelihood estimate converges in probability to the true covariance matrix $\boldsymbol{\Sigma}$.

Letting $n \to \infty$, we can write the argument of logarithm in (A.6) as

$$\left(\frac{|\boldsymbol{\Sigma}_{fa^*\cup A}|}{|\boldsymbol{\Sigma}_{mb^*\cup A}|}\right) \Big/ \left(\frac{|\boldsymbol{\Sigma}_{fa\cup A}|}{|\boldsymbol{\Sigma}_{mb\cup A}|}\right) \tag{A.7}$$

We can simplify the numerator and denominator by noticing that $\boldsymbol{\Sigma}_{fa^*\cup A}$ can be partitioned as

$$\begin{pmatrix} \text{var}(x_j) & \text{cov}(x_j, \boldsymbol{x}_{mb^*\cup A}) \\ \text{cov}(x_j, \boldsymbol{x}_{mb^*\cup A})^T & \boldsymbol{\Sigma}_{mb^*\cup A} \end{pmatrix},$$

where $\text{var}(x_j)$ is the variance of variable x_j, $\text{cov}(x_j, \boldsymbol{x}_{mb^*\cup A})$ is a horizontal vector containing covariances between x_j and each of the variables in set $mb^* \cup A$. Using basic results concerning determinants of a partitioned matrix (see, for instance, [42]), we have

$$\begin{aligned} |\boldsymbol{\Sigma}_{fa^*\cup A}| &= |\boldsymbol{\Sigma}_{mb^*\cup A}| \cdot (\text{var}(x_j) - \text{cov}(x_j,\ \boldsymbol{x}_{mb^*\cup A})\,(\boldsymbol{\Sigma}_{mb^*\cup A})^{-1}\,\text{cov}(x_j,\ \boldsymbol{x}_{mb^*\cup A})^T) \\ &= |\boldsymbol{\Sigma}_{mb^*\cup A}| \cdot \Big(\text{var}(x_j) - \text{var}(\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}])\Big) \\ &= |\boldsymbol{\Sigma}_{mb^*\cup A}| \cdot \text{var}\Big(x_j \mid \boldsymbol{x}_{mb^*\cup A}\Big), \end{aligned}$$

where $\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}] = \text{cov}(x_j,\ \boldsymbol{x}_{mb^*\cup A})\,(\boldsymbol{\Sigma}_{mb^*\cup A})^{-1}\,\boldsymbol{x}_{mb^*\cup A}$, which is the linear least squares predictor of x_j from $\boldsymbol{x}_{mb^*\cup A}$. The last equality follows from the definition of partial variance, which is the residual variance of x_j after subtracting the variance based on linear least squares predictor $\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}]$. Using this, we get

$$\frac{|\boldsymbol{\Sigma}_{fa^*\cup A}|}{|\boldsymbol{\Sigma}_{mb^*\cup A}|} = \text{var}\Big(x_j \mid \boldsymbol{x}_{mb^*\cup A}\Big).$$

Applying this also for the ratio of $|\boldsymbol{\Sigma}_{fa\cup A}|$ and $|\boldsymbol{\Sigma}_{mb\cup A}|$, lets us to write (A.7) as

$$\left(\frac{|\boldsymbol{\Sigma}_{fa^*\cup A}|}{|\boldsymbol{\Sigma}_{mb^*\cup A}|}\right) \Big/ \left(\frac{|\boldsymbol{\Sigma}_{fa\cup A}|}{|\boldsymbol{\Sigma}_{mb\cup A}|}\right) = \frac{\text{var}\Big(x_j \mid \boldsymbol{x}_{mb^*\cup A}\Big)}{\text{var}\Big(x_j \mid \boldsymbol{x}_{mb\cup A}\Big)}. \tag{A.8}$$

The form (A.8) makes it easier to analyse the behaviour of the determinant term and we can write the log ratio in (A.4) as follows

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*\cup A})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb\cup A})} = \frac{a}{2}\log n - \frac{n}{2}\log \frac{\text{var}\Big(x_j \mid \boldsymbol{x}_{mb^*\cup A}\Big)}{\text{var}\Big(x_j \mid \boldsymbol{x}_{mb\cup A}\Big)} + O_p(1). \tag{A.9}$$

By investigating (A.9), it is clear that consistency is achieved if we can show that

$$\frac{\text{var}\Big(x_j \mid \boldsymbol{x}_{mb^*\cup A}\Big)}{\text{var}\Big(x_j \mid \boldsymbol{x}_{mb\cup A}\Big)} < 1. \tag{A.10}$$

The equation (A.10) is equivalent to

$$\begin{aligned} &\text{var}\Big(x_j \mid \boldsymbol{x}_{mb^*\cup A}\Big) < \text{var}\Big(x_j \mid \boldsymbol{x}_{mb\cup A}\Big) \\ \Leftrightarrow\ &\text{var}(x_j) - \text{var}(\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}]) < \text{var}(x_j) - \text{var}(\hat{x}_j[\boldsymbol{x}_{mb\cup A}]) \\ \Leftrightarrow\ &\text{var}(\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}]) > \text{var}(\hat{x}_j[\boldsymbol{x}_{mb\cup A}]). \end{aligned} \tag{A.11}$$

Now assume $mb \neq \emptyset$, and denote the missing true Markov blanket members by $R = mb^* \backslash mb$. Then by using the additivity of the explained variance (see [2], p.138), we can write the left side of (A.11) as

$$\begin{aligned} \text{var}(\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}]) &= \text{var}(\hat{x}_j[\boldsymbol{x}_{mb\cup A\cup R}]) \\ &= \text{var}(\hat{x}_j[\boldsymbol{x}_{mb\cup A}]) + \text{var}(\hat{x}_j[\boldsymbol{x}_R - \hat{\boldsymbol{x}}_R[\boldsymbol{x}_{mb\cup A}]]). \end{aligned}$$

The term $\text{var}(\hat{x}_j[\boldsymbol{x}_R - \hat{x}_R[\boldsymbol{x}_{mb\cup A}]]) > 0$, since elements of R are in $x_j's$ Markov blanket. This shows that (A.10) holds.

If $mb = \emptyset$, the inequality (A.11) can be written as

$$\text{var}(\hat{x}_j[\boldsymbol{x}_{mb^*\cup A}]) > \text{var}(\hat{x}_j[\boldsymbol{x}_A]).$$

Using again the additivity of the explained variance, this becomes

$$\text{var}(\hat{x}_j[\boldsymbol{x}_A]) + \text{var}(\hat{x}_j[\boldsymbol{x}_{mb^*} - \hat{x}_{mb^*}[\boldsymbol{x}_A]]) > \text{var}(\hat{x}_j[\boldsymbol{x}_A]),$$

which clearly holds.

All in all, we have showed that

$$-\frac{n}{2}\log\frac{\text{var}\left(x_j \mid \boldsymbol{x}_{mb^*\cup A}\right)}{\text{var}\left(x_j \mid \boldsymbol{x}_{mb\cup A}\right)} \to \infty,$$

in probability, as $n \to \infty$. This implies that

$$\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb^*\cup A})}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb\cup A})} \to \infty$$

in probability, as $n \to \infty$, since n increases faster than $(a/2)\log n$ decreases.

Appendix A.3. Proof of Lemma 3

We want to show that asymptotically, adding a true edge into the Markov blanket always increases the score. Denote $|B| = p_j$ and $a = |A| - p_j$ in order to have consistent notation with the previous proofs.

Here $a = -1$, since the sets under consideration differ only by one node. Now, the analysis proceeds exactly as in the underestimation part of the global consistency proof, and we can write the log ratio as

$$\log\frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_B)}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_A)} = \frac{a}{2}\log n - \frac{n}{2}\log\frac{\text{var}\left(x_j \mid \boldsymbol{x}_B\right)}{\text{var}\left(x_j \mid \boldsymbol{x}_A\right)} + O_p(1).$$

To prove the claim, it suffices to show that

$$\frac{\text{var}\left(x_j \mid \boldsymbol{x}_B\right)}{\text{var}\left(x_j \mid \boldsymbol{x}_A\right)} < 1. \tag{A.12}$$

The equation (A.12) is equivalent to

$$\begin{aligned} &\text{var}\left(x_j \mid \boldsymbol{x}_B\right) < \text{var}\left(x_j \mid \boldsymbol{x}_A\right) \\ \Leftrightarrow\ &\text{var}(x_j) - \text{var}(\hat{x}_j[\boldsymbol{x}_B]) < \text{var}(x_j) - \text{var}(\hat{x}_j[\boldsymbol{x}_A]) \\ \Leftrightarrow\ &\text{var}(\hat{x}_j[\boldsymbol{x}_B]) > \text{var}(\hat{x}_j[\boldsymbol{x}_A]). \end{aligned} \tag{A.13}$$

By using the additivity of the explained variance (see [2], p.138), we can write the left side of (A.13) as

$$\begin{aligned} \text{var}(\hat{x}_j[\boldsymbol{x}_B]) &= \text{var}(\hat{x}_j[\boldsymbol{x}_{A\cup\{i\}}]) \\ &= \text{var}(\hat{x}_j[\boldsymbol{x}_A]) + \text{var}(\hat{x}_j[x_i - \hat{x}_i[\boldsymbol{x}_A]]). \end{aligned}$$

The term $\text{var}(\hat{x}_j[x_i - \hat{x}_i[\boldsymbol{x}_A]]) > 0$, since we assumed that node i is in the Markov blanket of j. This shows that (A.12) holds. If $A = \emptyset$, we have that $\hat{x}_j[\boldsymbol{x}_A] = 0$ and $\hat{x}_i[\boldsymbol{x}_A] = 0$. This lets us to write the inequality (A.13) as $\text{var}(\hat{x}_j[x_i]) > 0$, which is again satisfied due to the Markov blanket assumption.

Appendix A.4. Proof of Lemma 4

We can proceed as in the overestimation part of the global consistency proof. Since the sets A and B differ only by one node i, the log-ratio takes the following form

$$\log \frac{p(\boldsymbol{X}_j \mid \boldsymbol{X}_B)}{p(\boldsymbol{X}_j \mid \boldsymbol{X}_A)} = \frac{1}{2} \log n - \frac{n}{2} \log \left(\frac{|\boldsymbol{S}_{B\cup\{j\}}||\boldsymbol{S}_A|}{|\boldsymbol{S}_B||\boldsymbol{S}_{A\cup\{j\}}|} \right) + O_p(1).$$

Now, it holds that $x_j \perp\!\!\!\perp x_i \mid \boldsymbol{x}_B$, since $mb^* \subset B$, and by definition conditioning $\boldsymbol{x}_j$ on its Markov blanket renders it independent of all the remaining nodes.

As before, we can now notice that the determinant term is just the deviance, $\text{dev}(x_j \perp\!\!\!\perp x_i \mid \boldsymbol{x}_B)$, multiplied by a constant. Since the deviance has an asymptotic chi-squared distribution, the determinant term is bounded in probability. Thus, the asymptotic behaviour of the log-ratio is dominated by the term $(1/2) \log n$, which diverges to positive infinity as claimed.

Appendix B. Additional Numerical Results

Table B.3 contains results for all the considered dimensions in the structure learning tests with synthetic data.

p	n	OR		AND		HC		glasso		NBS		space	
		TP	FP	TP	FP	TP	FP	TP	FP	TP	FP	TP	FP
64	125	0.60	6e-03	0.44	9e-04	0.54	2e-03	0.53	1e-02	0.37	4e-04	0.66	2e-02
	250	0.72	3e-03	0.59	4e-04	0.68	1e-03	0.74	2e-02	0.57	6e-04	0.79	2e-02
	500	0.81	2e-03	0.73	2e-04	0.78	6e-04	0.86	2e-02	0.71	6e-04	0.88	2e-02
	1000	0.88	1e-03	0.83	1e-04	0.87	4e-04	0.93	3e-02	0.82	9e-04	0.94	2e-02
	2000	0.95	8e-04	0.91	6e-05	0.94	2e-04	0.97	4e-02	0.90	9e-04	0.98	2e-02
	4000	0.98	4e-04	0.96	4e-05	0.98	1e-04	0.99	4e-02	0.95	8e-04	0.99	2e-02
128	125	0.58	5e-03	0.41	8e-04	0.53	2e-03	0.36	3e-03	0.31	1e-04	0.61	6e-03
	250	0.71	3e-03	0.58	4e-04	0.67	1e-03	0.72	8e-03	0.52	2e-04	0.77	9e-03
	500	0.81	2e-03	0.72	2e-04	0.78	5e-04	0.85	1e-02	0.68	1e-04	0.87	1e-02
	1000	0.88	1e-03	0.83	1e-04	0.87	3e-04	0.91	2e-02	0.81	2e-04	0.93	1e-02
	2000	0.94	6e-04	0.91	6e-05	0.93	1e-04	0.93	9e-03	0.89	2e-04	0.97	9e-03
	4000	0.98	4e-04	0.96	6e-05	0.97	9e-05	0.97	1e-02	0.94	3e-04	0.99	9e-03
256	125	0.53	4e-03	0.38	7e-04	0.48	2e-03	0.05	8e-05	0.22	1e-05	0.49	1e-03
	250	0.68	2e-03	0.56	4e-04	0.64	9e-04	0.52	1e-03	0.44	3e-05	0.68	2e-03
	500	0.79	2e-03	0.70	2e-04	0.76	5e-04	0.71	2e-03	0.62	3e-05	0.82	4e-03
	1000	0.88	1e-03	0.82	1e-04	0.85	3e-04	0.84	2e-03	0.76	5e-05	0.91	5e-03
	2000	0.94	6e-04	0.90	8e-05	0.92	2e-04	0.92	3e-03	0.86	6e-05	0.96	6e-03
	4000	0.98	4e-04	0.96	5e-05	0.97	1e-04	0.97	4e-03	0.93	7e-05	0.99	5e-03
512	125	0.49	3e-03	0.35	6e-04	0.44	1e-03	0.01	0	0.15	4e-06	0.43	6e-04
	250	0.65	2e-03	0.53	3e-04	0.61	8e-04	0.48	3e-04	0.37	9e-06	0.61	6e-04
	500	0.77	1e-03	0.69	2e-04	0.74	5e-04	0.69	5e-04	0.57	1e-05	0.76	9e-04
	1000	0.86	8e-04	0.80	1e-04	0.84	3e-04	0.82	8e-04	0.73	1e-05	0.86	1e-03
	2000	0.93	5e-04	0.89	7e-05	0.92	2e-04	0.91	1e-03	0.83	1e-05	0.93	1e-03
	4000	0.97	4e-04	0.95	4e-05	0.97	1e-04	0.97	2e-03	0.91	1e-05	0.97	1e-03
1024	125	0.43	3e-03	0.29	5e-04	0.37	1e-03	0.00	0	0.08	7e-07	0.36	4e-04
	250	0.61	2e-03	0.49	3e-04	0.57	7e-04	0.40	7e-05	0.29	1e-06	0.56	3e-04
	500	0.74	1e-03	0.66	2e-04	0.72	4e-04	0.66	2e-04	0.51	2e-06	0.72	3e-04
	1000	0.84	7e-04	0.79	1e-04	0.82	3e-04	0.81	3e-04	0.68	3e-06	0.83	3e-04
	2000	0.92	5e-04	0.88	7e-05	0.90	2e-04	0.91	6e-04	0.81	2e-06	0.91	3e-04
	4000	0.97	3e-04	0.94	5e-05	0.96	1e-04	0.96	1e-03	0.89	3e-06	0.96	3e-04

Table B.3: A table showing true positive (TP) and false positive (FP) rates in structure learning tests for different methods and sample sizes.

In addition to true positive and false positive rates, we calculated the Matthews correlation coefficient (MCC) which can be obtained via the following formula (see, for instance, [43])

$$\mathrm{MCC} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{B.1}$$

where TP, FP, TN and FN denote the numbers of true positives, false positives, true negatives and false negatives, respectively.

When computing the MCCs, we used the average values for quantities appearing in (B.1) that are easily obtained from the numbers in Table B.3, as the true number of edges (and missing edges) in the generating model is known. The results are shown in Table B.4. The way the methods compare to each other follows mainly a similar pattern as seen with the Hamming distances in Figure 2. One difference can be seen with `glasso` when $d = 512$ or $d = 1024$ and $n = 125$. In these cases, the output of `glasso` is, on average, nearly an empty graph which is relatively close to the true generating graph when measured using Hamming distance. However, the corresponding MCC is close to zero and substantially smaller compared to other methods.

Average running times for the different methods in the structure learning tests are presented in Table B.5. For the marginal pseudo-likelihood methods and `NBS`, the shown times are average values computed from 10 tests. Note, that the result shown for the `HC`-method is the time it took to perform the hill-climb after the `OR`-graph was first estimated. The sparsity promoting prior was used in the tests. The column `maxMB` contains the maximum time on average taken by a single Markov blanket search which demonstrates the effect of parallelization.

p	n	OR	AND	HC	glasso	NBS	space
64	125	0.69	0.64	0.70	0.58	0.59	0.63
	250	0.80	0.76	0.80	0.67	0.74	0.71
	500	0.88	0.84	0.87	0.72	0.83	0.77
	1000	0.92	0.91	0.92	0.71	0.89	0.79
	2000	0.96	0.95	0.96	0.70	0.94	0.82
	4000	0.98	0.98	0.99	0.67	0.96	0.83
128	125	0.62	0.61	0.67	0.50	0.55	0.63
	250	0.76	0.75	0.79	0.67	0.71	0.68
	500	0.85	0.84	0.87	0.68	0.82	0.74
	1000	0.91	0.91	0.92	0.69	0.89	0.77
	2000	0.95	0.95	0.96	0.78	0.93	0.80
	4000	0.98	0.98	0.98	0.80	0.96	0.82
256	125	0.54	0.56	0.59	0.22	0.47	0.61
	250	0.70	0.72	0.74	0.65	0.66	0.70
	500	0.81	0.82	0.84	0.76	0.79	0.72
	1000	0.89	0.90	0.91	0.81	0.87	0.75
	2000	0.93	0.94	0.95	0.83	0.93	0.76
	4000	0.97	0.98	0.98	0.84	0.96	0.81
512	125	0.45	0.51	0.51	0.08	0.39	0.58
	250	0.62	0.68	0.69	0.65	0.60	0.71
	500	0.75	0.80	0.81	0.77	0.76	0.78
	1000	0.85	0.88	0.89	0.83	0.85	0.83
	2000	0.91	0.93	0.94	0.85	0.91	0.85
	4000	0.95	0.97	0.97	0.84	0.95	0.87
1024	125	0.34	0.42	0.41	0.03	0.29	0.50
	250	0.53	0.63	0.61	0.61	0.53	0.67
	500	0.68	0.77	0.76	0.77	0.71	0.78
	1000	0.79	0.86	0.85	0.83	0.83	0.85
	2000	0.87	0.92	0.92	0.84	0.90	0.90
	4000	0.92	0.96	0.96	0.81	0.95	0.93

Table B.4: A table showing the Matthews correlation coefficients in structure learning tests for different methods and sample sizes.

d	n	OR/AND	maxMB	HC	glasso	NBS	space
64	250	0.429	(0.021)	(0.054)	0.021	0.002	0.044
	1000	0.543	(0.023)	(0.079)	0.014	0.002	0.297
	4000	0.558	(0.025)	(0.082)	0.009	0.003	1.359
128	250	1.832	(0.042)	(0.108)	0.134	0.008	0.161
	1000	1.908	(0.045)	(0.138)	0.089	0.010	1.078
	4000	2.076	(0.047)	(0.164)	0.054	0.007	6.715
256	250	7.059	(0.074)	(0.231)	1.184	0.043	0.761
	1000	7.727	(0.084)	(0.293)	0.656	0.044	3.904
	4000	8.284	(0.089)	(0.335)	0.423	0.044	35.831
512	250	29.448	(0.153)	(0.559)	11.853	0.326	3.899
	1000	31.317	(0.177)	(0.627)	5.254	0.336	21.640
	4000	33.505	(0.179)	(0.711)	3.305	0.330	187.473

Table B.5: Average running times in seconds for different methods in structure learning tests.

In each of the ten tests (with given sample size and dimension), space and glasso were computed using 12 different values for the tuning parameters as explained in the paper. Shown results for these two methods are averages computed over different tests and also over different tuning parameter values. All the timing experiments were run in Matlab or R on a standard laptop with a 2.30 GHz quad-core processor.

Figure B.6 presents results for the prediction tests using synthetic data. In these experiments, a data set of 2048 observations from the same model structures as used in the structure learning tests was created and the procedure used with the brain data was repeated. Size of the training set was 2000 and the remaining 48 observations formed the test set. Selection of tuning parameters for glasso, space and NBS was done as in the structure learning tests. Figure B.6 shows the MSE and the corresponding number of edges in the graphical model for dimensions 64 and 128. The shown results are averages computed from 25 data sets. Here, all the fractional marginal pseudo-likelihood based methods have slightly better prediction performances compared to other methods.

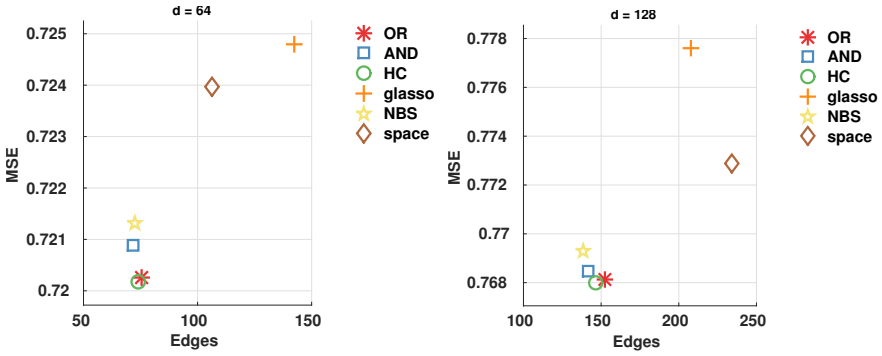

Figure B.6: MSE (vertical axis) and the number of found edges (horizontal axis) for the synthetic data.

Appendix C. Pseudocode for the Search Algorithms

The Algorithm 1 describes the procedure discussed in Section 3.4 for finding the Markov blankets for each node using the fractional marginal pseudo-likelihood as scoring function. The Algorithm 2 is the greedy hill-climbing procedure in the reduced model space used by the `HC`-method.

Algorithm 1 Procedure for optimizing the local fractional marginal pseudo-likelihood of a node j.

procedure MARKOV-BLANKET-HILL-CLIMB(j, $\boldsymbol{X}$)
  $mb(j), \widehat{mb}(j) \leftarrow \varnothing$
  while $\widehat{mb}(j)$ has changed **do**
    $C \leftarrow V \setminus (mb(j) \cup \{j\})$
    $mb(j) \leftarrow \widehat{mb}(j)$
    for $i \in C$ **do**
      if $\log p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)\cup\{i\}}) > \log p(\boldsymbol{X}_j \mid \boldsymbol{X}_{\widehat{mb}(j)})$ **then**
        $\widehat{mb}(j) \leftarrow mb(j) \cup \{i\}$
      end if
    end for
    while $\widehat{mb}(j)$ has changed **do**
      $mb(j) \leftarrow \widehat{mb}(j)$
      for $i \in mb(j)$ **do**
        if $\log p(\boldsymbol{X}_j \mid \boldsymbol{X}_{mb(j)\setminus\{i\}}) > \log p(\boldsymbol{X}_j \mid \boldsymbol{X}_{\widehat{mb}(j)})$ **then**
          $\widehat{mb}(j) \leftarrow mb(j) \setminus \{i\}$
        end if
      end for
    end while
  end while
  return $\widehat{mb}(j)$
end procedure

Algorithm 2 Procedure for optimizing the fractional marginal pseudo-likelihood.

procedure GRAPH-HILL-CLIMB($\mathcal{G}_{\mathtt{DR}}$,$\boldsymbol{X}$)
  $G, \widehat{G} \leftarrow$ *empty graph*
  while $\widehat{G}$ has changed **do**
    $G \leftarrow \widehat{G}$
    for $G' \in N_{\mathcal{G}_{\mathtt{DR}}}(G)$ **do**
      if $\hat{p}(\boldsymbol{X} \mid G') > \hat{p}(\boldsymbol{X} \mid \widehat{G})$ **then**
        $\widehat{G} \leftarrow G'$
      end if
    end for
  end while
  return $\widehat{G}$
end procedure